Quantifier-free logic for multialgebraic theories

Yngve Lamo* & MichałWalicki†

March 5, 2003

Abstract

We develop a new logic for deriving consequences of multialgebraic theories (specifications). Multilagebras are used as models for nondeterminism in the context of algebraic specifications. They are many sorted algebras with *set valued* operations. Atomic formulae are set inclusion $t \prec t'$ – the interpretation of t is included in the interpretation of t', and element equality $t \doteq t'$ – t and t' denote the same element of the carrier. We introduce the Rasiowa-Sikorski logic R-S for proving multilagebraic tautologies and show its soundness and completeness. We then extend this system for proving consequences of specifications based on translation of theories into logical formulae. Finally, we show how such a translation may be avoided – introduction of *specific cut* rules leads to a sound and complete Gentzen system for proving directly consequences of specifications.

Introduction

The institution of multialgebras, $\mathcal{MA}$, [7], provides a powerful algebraic framework for specification – primarily, but not exclusively, of nondeterministic behavior [3, 14, 7]. In a multialgebra a nondeterministic operation returns the set of all possible outcomes for the operation. Hence operations are interpreted as functions from the carrier to the powerset of the carrier. The particular case of the empty result set gives straightforwardly a subinstitution of partial algebras [6, 8]. The logic has two atoms: set inclusion, $\prec$ and element equality $\doteq$. The set inclusion $t \prec t'$ holds iff the interpretation of t is included in the interpretation of t', i.e. every possible value for t is a possible value for t'. In other words: the term t is not more nondeterministic than t'. The element equality $t \doteq t'$ states that the terms t and t' must return the same element. In other words: t and t' are deterministic.

Formulae used for writing specifications are sequents over atomic equalities and inclusions. Our objective is to design a quantifier-free logic for deriving consequences of such specifications. First, using the technique of Rasiowa-Sikorski from [11], we design a sound and complete system R-S. This system could be seen as a sublogic of the first order logic for multialgebras given by Konikowska and Białasik in [2]. However, their language does not include the element equality $\doteq$. This predicate can't be expressed in their language by a set of formulae without the use of quantifiers. This is also related to the fact that to express emptiness or non-emptiness of the carrier, quantified formulae are needed. E.g., $\exists x : x \prec x$ expresses non-emptiness of the carrier which, in our language, can be expressed by the quantifier-free formula $x \doteq x$ (with *only implict* quantification over possible assignments). Finally, and most significantly, the language from [2], unless extended to full first-order, is not expressive enough to state non-emptiness of any result set. Consequently, even the quantifier free tautologies have all to take into account the possibility that any involved term may yield an empty result.

This not only yields fewer and less specific tautologies, but has also more practical aspects. Writing specifications one certainly wants the possibility to state that a term is deterministic. The axiom $f(x) \doteq f(x)$ states that the operation f is a total function, and such statements figure naturally as assumptions (or consequences) in the formulae one wants to prove – preferably without the use of full first-order logic. The corresponding formula in the language from [2] would be $\exists y : y \prec f(x) \wedge f(x) \prec y$. Besides, there is the whole tradition of algebraic specifications based

*Høgskolen i Bergen, email: yla@hib.no
†Universitetet i Bergen, email: michal@ii.uib.no

on equational axioms and equational reasoning. The element equality, present in the institution of multialgebras, [7], makes comparison and embedding of other institutions to the institution of multialgebras simple and straightforward, without the use of quantifiers.

Although we consider the lack of a connective corresponding to $\doteq$ in [2] a serious drawback, our development and presentation owe quite a lot to this work. We utilize the technique of Rasiowa and Sikorski, [11], which was brought to our attention by [2] and which is nicely summarized in [4, 5] (and recently used also in [1]). It gives a general way for designing logics based on the semantic properties of the atomic predicates, and we apply it to our case of multialgebraic specifications with $\prec$ and $\doteq$.

Having introduced the relevant and basic notions from multialgebras in Section 1, we design a Rasiowa-Sikorski system, R-S, for quantifier-free logic over $\prec$ and $\doteq$ in Section 2, which includes also proofs of soundness and completeness. Following the cited works, we also define a unique deduction strategy which can be used for implementing the logic. These two section repeat exactly the respective sections from the earlier version [9]. The changes wrt. [9] concern the rest of the report devoted to treatement of axioms and consequences of specifications. In Section 3, we address the issue of proving consequences of specifications. Specifications are sets of sequents and we want to derive their consequences, i.e., new sequents. We indicate the required translation schema and extend the R-S system with one rule needed for this purpose. Finally, in Section 4, we transform the obtained system to a sound and complete Gentzen calculus GS, which is more user-friendly than the R-S system for proving theorems by hand. In order to handle proofs of consequences of theories without any intermediary translation of the involved sequents, we replace the axiom rule (as well as various rules for the logical connectives) by the specific cut rules, originating from [10]. We thus obtain a system for direct reasoning about specification, where reasoning about sequents over atomic formulae involves only such sequents. Besides extention of the language with the useful predicate $\doteq$, we consider this result an improvement – by simplification – of the full first-order Gentzen system from [2].

1 Multialgebras

We use standard algebraic signatures to present multialgebra specifications:

Definition 1.1 *A signature Σ is a pair $\Sigma = (S, \Omega)$, where S is a set of sort names, and $\Omega = \{\Omega_{s^*,s} : s^* \in S^*, s \in S\}$ is a collection of sets of operation names. We write $(f : s^* \to s) \in \Omega$ to denote that $f \in \Omega_{s^*,s}$. If s^* is the empty string then f is called a constant of sort s.*

Ground terms, $\mathcal{T}_\Sigma$, and terms over a given set X of variables, $\mathcal{T}_{\Sigma,X}$, over a signature Σ, are defined in the usual way.

Terms and variables are sorted by sort names from the signature. We write t_s and x_s to indicate that the term t and the variable x has sort s. We assume that any term is well sorted so we write t, x without sorting when the sorting is implicit.

Definition 1.2 *The well formed formulae over a signature Σ and variables X is the least set $\mathcal{F}_{\Sigma,X}$ such that:*

- *if $t_s, t'_s \in \mathcal{T}_{\Sigma,X}$, then: $t_s \prec t'_s, t_s \doteq t'_s \in \mathcal{F}_{\Sigma,X}$ – these are atomic formulae*
- *if $\gamma, \phi \in \mathcal{F}_{\Sigma,X}$, then $\neg\gamma, \gamma \vee \phi, \gamma \wedge \phi \in \mathcal{F}_{\Sigma,X}$*

The implication sign $\to$ can be introduced in the usual way: $\phi \to \psi \iff \neg\phi \vee \psi$.

Definition 1.3 *A Σ multialgebra A for a signature Σ is a pair $A = (S^A, \Omega^A)$, where $S^A = \{s^A : s \in S\}$ is a carrier set for each sort name $s \in S$, and $\Omega^A = \{f^A : f \in \Omega\}$ is a set valued function for each operation name $f \in \Omega$, such that for each $f : s_1 \times \ldots \times s_n \to s$ we have:*

$$f^A : s_1^A \times \ldots s_n^A \to \mathcal{P}(s^A)$$

where $\mathcal{P}(s^A)$ is the set of all subset of s^A.

Composition of operations is defined by pointwise extension, i.e., $f^A(g^A(x)) = \bigcup_{y \in g^A(x)} f^A(y)$.

The carrier set of a multialgebra A is denoted by $|A|$ and the carrier set of the sort s is denoted by $|A|_s$. Note that a carrier set can be empty for some sorts. An operation is called partial if it returns the empty set for some arguments. An operation returning more than one value for some arguments is called nondeterministic. An operation that is neither partial nor nondeterministic is a function. In other words a function is a total deterministic operation.

Definition 1.4 *Given a multialgebra A, an assignment α is a function $\alpha : X \to |A| \uplus \{\emptyset\}$ where $\alpha(x_s) = \emptyset \iff |A|_s = \emptyset$.*

So an assignment assigns an element of the carrier – not a *set* of elements! – to each variable of a nonempty sort.[1]

Definition 1.5 *A Σ structure $M = \langle A, \alpha \rangle$ is a Σ multialgebra A, together with an assignment α.*

Given a structure M, all terms $t \in \mathcal{T}_{\Sigma,X}$ obtain a unique interpretation, denoted by t^M, which is defined in the standard way.

Definition 1.6 *Let $M = \langle A, \alpha \rangle$ be a Σ structure. The satisfaction relation $\models$ is defined by:*

1. $M \models t \prec t' \iff t^M \subseteq t'^M$
2. $M \models t \doteq t' \iff t^M = \{e\} = t'^M$, *where e is an element of the carrier (we do not usually distinguish one-element set $\{e\}$ and the element e.)*
3. $M \models \neg\gamma \iff M \not\models \gamma$
4. $M \models \gamma \vee \phi \iff M \models \gamma$ *or* $M \models \phi$
5. $M \models \gamma \wedge \phi \iff M \models \gamma$ *and* $M \models \phi$

Remark 1.7 *According to point 2, an equality may hold only if the carrier is non-empty. Given a structure $M = \langle A, \alpha \rangle$, we have that:*

- $M \models \neg(x_s \doteq x_s) \iff |A|_s = \emptyset$
- $M \models x_s \doteq x_s \iff |A|_s \neq \emptyset$

Also, if $|A|_s = \emptyset$, we have for any terms t_s, t'_s:

- $M \models t_s \prec t'_s$ *and*
- $M \not\models \neg(t_s \prec t'_s)$

We introduce logical symbols, abbreviating the formulae stating that a carrier is empty or not.

Definition 1.8 *We define the symbols $\mathcal{E}_s \equiv \neg(x_s \doteq x_s)$, for any $x_s \in X_s$, and $\neg\mathcal{E}_s \equiv x_s \doteq x_s$, for any $x_s \in X_s$. By remark 1.7, for any structure $M = \langle A, \alpha \rangle$:*

- $M \models \mathcal{E}_s \iff |A|_s = \emptyset$
- $M \models \neg\mathcal{E}_s \iff |A|_s \neq \emptyset$

2 The R-S calculus

We now present a quantifier free Rasiowa-Sikorski (R-S) deduction system with set inclusion and equality for multialgebras. The R-S system illustrates a powerful way of designing logical deduction systems based on semantical properties of atomic predicates of the language, which was originally introduced in [11]. Our presentation in this section is an adaptation and extension of a similar logic described in [2].

The system processes sets of formulae (clauses). However, it allows one also to define a specific deduction strategy in which such sets are considered as ordered *sequences* of formulae without repetitions – this is the interpretation we will be using in this and next section. Particular sequences are singled out as *axiomatic*, in our case, sequences containing a formula or subsequence of the form:

- $x \prec x$ for a variable x
- $\phi, \neg\phi$ for a formula ϕ
- $\neg\mathcal{E}_s, t_s \prec t'_s$.

The order of occurrences of such formulae in a sequence does not matter. Given a set of formulae, we say that it satisfies the 'axiomatic sequence condition' if the set involves formulae from which an axiomatic sequence could be formed.

[1] One could alternatively define assignment as a partial function, $\alpha : X \to |A|$, with domain being the variables over nonempty sorts. Our formulation gives immediately that any non-ground term with variables from empty sort will be empty, since operations in multialgebra applied to empty set yield empty set.

Definition 2.1 *A structure $M = \langle A, \alpha \rangle$ satisfies a sequence $\Gamma = \gamma_1, \ldots, \gamma_n$, written $M \models \Gamma$, iff $M \models \gamma_i$ for some i.*

Hence the "," should be viewed as a meta disjunction.

An R-S rule has one of the following forms, where Γ_i are sequences:

$$\frac{\Gamma_1}{\Gamma_2}, \quad \frac{\Gamma_1}{\Gamma_2 \mid \Gamma_3}, \text{ or } \quad \frac{\Gamma_1}{\Gamma_2 \mid \Gamma_3 \mid \Gamma_4}$$

Both sides of the " $|$ " sign have to hold for making an expression involving $|$ true, hence it should be viewed as a meta conjunction.

The rules are designed so that they are invertible and one uses a strong notion of soundness.

Definition 2.2 *An (R-S) rule is sound when, for any structure M, M satisfies the premise iff it satisfies the conclusion.*

The strength of this notion lies not only in the requirement of invertibility but also in the use of a structure – it says that the premise is satisfied iff the conclusion is *for any given* assignment. (The usual notion of soundness is, of course, implied by this one.)

In addition to axiomatic sequences, one also identifies the *indecomposable* sequences – in our case, these are given by the following definition.

Definition 2.3 *A Σ formula is indecomposable iff it has one of the following forms:*

- *$\mathcal{E}_s$ or $\neg\mathcal{E}_s$, $s \in S$*
- *$x \prec y$ or $\neg(x \prec y)$, $x, y \in X$*
- *$x \prec f(x_1, \ldots, x_n)$ or $\neg(x \prec f(x_1, \ldots, x_n))$, $x, x_i \in X$ and $f \in \Omega$ (f is possibly a constant).*

A sequence of formulae is indecomposable iff every formula in the sequence is indecomposable.

No rules can modify the indecomposable formulae and so if such a formula appears in a sequence during the proof, it will not be changed by any subsequent application of rules.

The R-S calculus has two types of rules, *replacement* rules and *expansion* rules. The goal of the replacement rules is to transform decomposable formulae leading either to axiomatic sequences or to indecomposable formulae (i.e., expressions involving only variables or function application to variables). Such rules have only one explicit formula in the premise sequence which is transformed, possibly with addition of a new formula, in the conclusion. There is exactly one decomposition rule for each case of a decomposable formula and schemata for decomposable formulae are disjoint, i.e., precisely one decomposition rule can be applied to any decomposable formula at any stage. In particular, we will have one rule for every positive decomposable formula, like $t \doteq t'$, and one rule for the corresponding negative formula, $\neg(t \doteq t')$.

The expansion rules are used to add logical consequences of the indecomposable formulae from the premise. They merely augment the premise sequence with some additional formulae without changing the formula itself.

In the notation for rules, we will use the sign "*" to indicate repetition of the active formula from the premise in the conclusion. The rule (VII–) given in the calculus below:

$$\frac{\Gamma', \neg(t \prec t'), \Gamma''}{\Gamma', x \prec t, \Gamma'', * \mid \Gamma', \neg(x \prec t'), \Gamma'', *} \quad \text{where } t \notin X \text{ and } x \in X \text{ arbitrary}$$

should be read in the following way

$$\frac{\Gamma', \neg(t \prec t'), \Gamma''}{\Gamma', x \prec t, \Gamma'', \neg(t \prec t') \text{ and } \Gamma', \neg(x \prec t'), \Gamma'', \neg(t \prec t')} \quad \text{where } t \notin X \text{ and } x \in X \text{ arbitrary}$$

Such a repetition will take place only in the implicit presence of existential quantifier and will be needed to ensure the possibility of finding an adequate witness. The above rule can be thus seen as:

$$\frac{\Gamma', \neg(t \prec t'), \Gamma''}{\exists x : \Gamma', x \prec t, \Gamma'', \text{ and } \Gamma', \neg(x \prec t'), \Gamma'',} \quad \text{where } t \notin X$$

We include all relevant rules and axioms from the logic given in [2] (i.e., except the replacement rules for quantifiers and the let-construction). We extended the logic by new replacement rules for the element equality, i.e. the rules (IX+) to (XI–).

Remember that the formula $\neg\mathcal{E}_s$ is a logical symbol abbreviating the statement that the carrier of a sort s is nonempty, i.e., $x_s \doteq x_s$. Similarly, $\mathcal{E}_s$ denotes that the carrier of sort s is empty, i.e., $\neg x_s \doteq x_s$. The only function of the restriction $t_s \neq x_s$ in the rules (X) and (XI) is to prevent further decomposition of such formulae.

The R-S proof system

Axiomatic sequences *(order does not matter)*

(I) $\Gamma, x \prec x, \Gamma'$ for $x \in X$ (II) $\Gamma, \gamma, \Gamma', \neg\gamma, \Gamma''$ for $\gamma \in \mathcal{F}_{\Sigma,X} \cup \{\mathcal{E}\}$ (III) $\Gamma, \neg\mathcal{E}_s, \Gamma', t_s \prec t'_s, \Gamma''$

Replacement rules *(unique decomposable premise formula)*

	+	−
(IV)		$\dfrac{\Gamma', \neg\neg\gamma, \Gamma''}{\Gamma', \gamma, \Gamma''}$
(V)	$\dfrac{\Gamma', \gamma \vee \phi, \Gamma''}{\Gamma', \gamma, \phi, \Gamma''}$	$\dfrac{\Gamma', \neg(\gamma \vee \phi), \Gamma''}{\Gamma', \neg\gamma, \Gamma'' \mid \Gamma', \neg\phi, \Gamma''}$
(VI)	$\dfrac{\Gamma', \gamma \wedge \phi, \Gamma''}{\Gamma', \gamma, \Gamma'' \mid \Gamma', \phi, \Gamma''}$	$\dfrac{\Gamma', \neg(\gamma \wedge \phi), \Gamma''}{\Gamma', \neg\gamma, \neg\phi, \Gamma''}$
(VII)	$\dfrac{\Gamma', t \prec t', \Gamma''}{\Gamma', \neg(x \prec t), x \prec t', \Gamma''}$ where $t \notin X$, and $x \in X$ is fresh	$\dfrac{\Gamma', \neg(t \prec t'), \Gamma''}{\Gamma', x \prec t, \Gamma'', * \mid \Gamma', \neg(x \prec t'), \Gamma'', *}$ where $t \notin X$ and $x \in X$ arbitrary
(VIII)	$\dfrac{\Gamma', x \prec f(\ldots, t, \ldots), \Gamma''}{\Gamma', y \prec t, \Gamma'', * \mid \Gamma', x \prec f(\ldots, y, \ldots), \Gamma'', *}$ where $y \in X$ arbitrary and $t \notin X$	$\dfrac{\Gamma', \neg(x \prec f(\ldots, t, \ldots)), \Gamma''}{\Gamma', \neg(y \prec t), \neg(x \prec f(\ldots, y, \ldots)), \Gamma''}$ where $y \in X$ is fresh and $t \notin X$
(IX)	$\dfrac{\Gamma', t \doteq t', \Gamma''}{\Gamma', t \doteq x, \Gamma'', * \mid \Gamma', t' \doteq x, \Gamma'', *}$ where $t, t' \notin X$ and $x \in X$ arbitrary	$\dfrac{\Gamma', \neg(t \doteq t'), \Gamma''}{\Gamma', \neg(t \doteq x), \neg(t' \doteq x), \Gamma''}$ where $t, t' \notin X$ and $x \in X$ is fresh
(X)	$\dfrac{\Gamma', t_s \doteq x_s, \Gamma''}{\Gamma', t_s \prec x_s, \Gamma'' \mid \Gamma', x_s \prec t_s, \Gamma'' \mid \Gamma', \neg\mathcal{E}_s, \Gamma''}$ where $x_s \in X$ and $t_s \neq x_s$	$\dfrac{\Gamma', \neg(t_s \doteq x_s), \Gamma''}{\Gamma', \mathcal{E}_s, \neg(x_s \prec t_s), \neg(t_s \prec x_s), \Gamma''}$ where $x_s \in X$ and $t_s \neq x_s$
(XI)	$\dfrac{\Gamma', x_s \doteq t_s, \Gamma''}{\Gamma', t_s \prec x_s, \Gamma'' \mid \Gamma', x_s \prec t_s, \Gamma'' \mid \Gamma', \neg\mathcal{E}_s, \Gamma''}$ where $x_s \in X$ and $t_s \neq x_s$	$\dfrac{\Gamma', \neg(x_s \doteq t_s), \Gamma''}{\Gamma', \mathcal{E}_s, \neg(x_s \prec t_s), \neg(t_s \prec x_s), \Gamma''}$ where $x_s \in X$ and $t_s \neq x_s$

Expansion rules *(indecomposable premise formulae)*

$$\text{(XII)} \quad \frac{\Gamma', \neg(x \prec y), \Gamma''}{\Gamma', \neg(x \prec y), \neg(y \prec x), \Gamma''}$$

$$\text{(XIII)} \quad \frac{\Gamma', \neg(y \prec x), \Gamma'', \neg(x \prec z), \Gamma'''}{\Gamma', \neg(y \prec x), \Gamma'', \neg(x \prec z), \neg(y \prec z), \Gamma'''}$$

$$\text{(XIV)}\quad \frac{\Gamma', \neg(y \prec x), \Gamma'', \neg(x \prec f(\overline{z})), \Gamma'''}{\Gamma', \neg(y \prec x), \Gamma'', \neg(x \prec f(\overline{z})), \neg(y \prec f(\overline{z})), \Gamma'''}\quad f \text{ is possibly a constant}$$

$$\text{(XV)}\quad \frac{\Gamma', \neg(y \prec z), \Gamma'', \neg(x \prec f(\ldots, z, \ldots)), \Gamma'''}{\Gamma', \neg(y \prec z), \Gamma'', \neg(x \prec f(\ldots, z, \ldots)), \neg(x \prec f(\ldots, y, \ldots)), \Gamma'''}\quad f \text{ is possibly a constant}$$

$$\text{(XVI)}\quad \frac{\Gamma', \neg\mathcal{E}_s, \Gamma'', \neg(x_{s'} \prec f(\ldots, y_s, \ldots)), \Gamma'''}{\Gamma', \neg\mathcal{E}_s, \Gamma'', \neg(x_{s'} \prec f(\ldots, y_s, \ldots)), \neg\mathcal{E}_{s'}, \Gamma'''}\quad f \text{ is possibly a constant}$$

Example 2.4 *We prove the tautology:* $c \doteq c, c \prec x \to x \prec c$, *i.e:* $\neg(c \doteq c), \neg(c \prec x), x \prec c$. *We mark the active formulae by boldface. Similarly, the variable introduced in the conclusion is in boldface. If a branch terminates, the axiomatic subsequences are underlined. We drop sort subscripts.*

$$\frac{\neg(\mathbf{c} \doteq \mathbf{c}), \neg(c \prec x), x \prec c}{\neg(c \doteq \mathbf{y}), \neg(c \prec x), x \prec c}\quad \textit{(IX–)}$$

$$\frac{\neg(\mathbf{c} \doteq \mathbf{y}), \neg(c \prec x), x \prec c}{\neg(y \prec c), \mathcal{E}, \neg(c \prec y), \neg(c \prec x), x \prec c}\quad \textit{(X–)}$$

$$\frac{\neg(y \prec c), \mathcal{E}, \neg(\mathbf{c} \prec \mathbf{y}), \neg(c \prec x), x \prec c}{\underline{\neg(y \prec c)}, \mathcal{E}, \underline{(\mathbf{y} \prec c)}, \neg(c \prec x), x \prec c, * \mid \neg(y \prec c), \mathcal{E}, \neg(\mathbf{y} \prec y), \neg(c \prec x), x \prec c, *}\quad \textit{(VII–)}$$

$$\frac{\neg(y \prec c), \mathcal{E}, \neg(y \prec y), \neg(\mathbf{c} \prec \mathbf{x}), x \prec c, \neg(c \prec y)}{\underline{\neg(y \prec c)}, \mathcal{E}, \neg(y \prec y), \underline{\mathbf{y} \prec c}, x \prec c, \neg(c \prec y), * \mid \neg(y \prec c), \mathcal{E}, \neg(y \prec y), \neg(\mathbf{y} \prec x), x \prec c, \neg(c \prec y), *}\quad \textit{(VII–)}$$

$$\frac{\neg(y \prec c), \mathcal{E}, \neg(y \prec y), \neg(\mathbf{y} \prec \mathbf{x}), x \prec c, \neg(c \prec y), \neg(c \prec x)}{\neg(y \prec c), \mathcal{E}, \neg(y \prec y), \neg(y \prec x), \neg(x \prec y), x \prec c, \neg(c \prec y), \neg(c \prec x)}\quad \textit{(XII)}$$

$$\frac{\neg(\mathbf{y} \prec \mathbf{c}), \mathcal{E}, \neg(y \prec y), \neg(y \prec x), \neg(\mathbf{x} \prec \mathbf{y}), x \prec c, \neg(c \prec y), \neg(c \prec x)}{\neg(y \prec c), \mathcal{E}, \neg(y \prec y), \neg(y \prec x), \neg(x \prec y), \underline{\neg(x \prec c)}, \underline{x \prec c}, \neg(c \prec y), \neg(c \prec x)}\quad \textit{(XIV)}$$

2.1 Construction of a unique deduction tree

Before proving soundness and completeness of the calculus, we show first that for a given sequence $\Gamma = \gamma_1, \ldots, \gamma_n$ one can choose a unique, canonical deduction tree. This fact will be of used in the proof of completeness, but it is also of independent importance since it suggests the way of possible implementation of the logic.

The strategy was illustrated in the above example 2.4. We start with the first formula γ_1. If it is decomposable, we apply the appropriate rule, (IX–). We now check whether the obtained indecomposable formulae ("to the left" of the "active position" in the obtained sequence[2]) can be used in any expansion rule and if they can, we apply the rule. This was not possible in the example, so we repeated the appllication of a decomposition rule in the second step. After this step, still no expansion rule was applicable to the obtained formulae $\neg(y \prec c), \mathcal{E}$ – so we consider the next formula to the right to which we could apply a decomposition rule, (VII–). Left branch gets closed and in the right we consider all indecomposable formulae obtained so far ("to the left" of the "active position"). Transitivity rule (XIV) applied to $\neg(y \prec c), \neg(y \prec y)$ yields a repetition, so it is not applied. The first possibility is application of the symmetry rule (XII) to $\neg(x \prec y)$, after which an application of (XIV) yields an axiomatic sequence. The following definition captures the above strategy.

Definition 2.5 *An R-S rule ρ is correctly applicable to a sequence Γ iff one of the following conditions is satisfied:*

1. *ρ is an R-S rule which augments Γ by some new formula or*
2. *there is no rule with the above property that can be applied to a formula or pair of formulae that lies to the left of the (active) formula or pair of formulae to which ρ is applicable.*

[2]Indexing formulae in a sequence as $\gamma_1, \gamma_2, \ldots, \gamma_n$, by "$\gamma_i$ lying to the left of γ_j" we mean simply that $i \leq j$. The "active position" is the index of the explicit formula from the premise – the one which has been processed by the rule.

The first point refers exclusively to the expansion rules. In the second point, ρ may be a replacement rule in which case it is applied to the leftmost decomposable formula, so that no expansion rule can be applied "to the left" of it. If, in this second case, ρ turns out to be an expansion rule, we see that first we have to apply the rule with one premise formula, i.e., (XII) or else the rule with two premise formulae which, together, lie as far "to the left" as possible. Since we only can use one replacement rule for a formula at any time and point 2 in 2.5 uniquely defines the expansion rule that is correctly applicable, we get that there is at most one R-S rule that is correctly applicable to any sequence Γ at any time.

By a deduction tree for a sequence Γ we mean a tree with Γ labelling the root, where the number and labelling of the children of each node originates from the application of some rule to the (sequence labeling the) node itself. Such a tree is a proof if all leaves are labeled by axiomatic sequences. The above definition and remarks allow us to define a unique decomposition tree for any sequence.[3] We identify vertex v with its label Π.

Definition 2.6 *A decomposition tree $DT(\Gamma)$ for a sequence Γ is a ternary tree with vertices being (or labelled by) sequences of formulae defined inductively by:*

1. *The root of $DT(\Gamma)$ is Γ.*
2. *If a vertex Π is either:*
   - *(a) an axiomatic sequence or*
   - *(b) an indecomposable sequence to which no expansion rule is correctly applicable*

   then Π is a leaf.
3. *Otherwise the vertex Π has:*
   - *(a) A single child Π', if the unique rule correctly applicable to Π has a single conclusion.*
   - *(b) Two children Π', Π'', if the unique rule correctly applicable to Π has two conclusions.*
   - *(c) Three children Π', Π'', Π''', if the unique rule correctly applicable to Π has three conclusions.*

We thus obtain

Lemma 2.7 *For any sequence Γ one can choose a unique decomposition tree, $DT(\Gamma)$.*

This fact can be used in implementation of the logic. We will use it in the proof of completeness, where also the following, obvious property will be of importance.

Lemma 2.8 *Assume that $B = \Pi_1, \Pi_2, \Pi_3, \ldots$ is an infinite branch in $DT(\Gamma)$ and let Γ_B be the set of all formulae occuring on the vertices of B. Then:*

1. *Γ_B is closed under all expansion rules.*
2. *If $\gamma_i \in \Gamma_B$ is decomposable, then there exists a vertex $\Pi_i \in B$ such that $\Pi_i = \Gamma', \gamma_i, \Gamma''$, where Γ' is indecomposable (possibly empty) and closed under all expansion rules. The vertex Π_{i+1}, following Π_i in B, is (one of) the conclusion(s) $\Gamma', \gamma_{i+1}, \Gamma''$ obtained by the correct application of the appropriate decomposition rule to Π_i.*

2.2 Soundness

Theorem 2.9 *The R-S system is sound.*

Proof. We only have to prove that the replacement rules involving $\doteq$ are sound, i.e. the rules (IX+) to (XI–). The axiomatic sequences and the remaining rules were proved sound in [2]. We consider an arbitrary structure $M = \langle A, \alpha \rangle$, and write $|M|$ for $|A|$. We drop sort subscript assuming that we always address only the relevant sort. We consider only the cases when a sequence is satisfied because the explicit (active) formulae are satisfied, since the other cases are trivial.

[3]We assume, in general, the the set X of all variables is countable. More generally, here we only need the assumption that it is well ordered, so that we can choose the first variable not present in a sequence for a fresh variable and we choose "the next variable in the ordering" for an arbitrary variable (for each formula which is processed repeatedly, i.e., inherited as indicated by *).

1. (IX+)

$$\frac{\Gamma', t \doteq t', \Gamma''}{\Gamma', t \doteq x, \Gamma'', * \mid \Gamma', t' \doteq x, \Gamma'', *} \quad \text{where } t, t' \notin X \text{ and } x \in X \text{ arbitrary}$$

$\Downarrow$ If $M \models t \doteq t'$, then M trivially satisfies both conclusions since $t \doteq t'$ is repeated on each side of the conjunction sign.
$\Uparrow$ If $M \models t \doteq x$ and $M \models t' \doteq x$, then $t^M = x^M = e = x^M = t'^M$, where $e \in |M|$, so $M \models t \doteq t'$.

2. (IX–)

$$\frac{\Gamma', \neg(t \doteq t'), \Gamma''}{\Gamma', \neg(t \doteq x), \neg(t' \doteq x), \Gamma''} \quad \text{where } t, t' \notin X \text{ and } x \in X \text{ is a new variable}$$

$\Downarrow$ When $M \models \neg(t \doteq t')$, we have two cases:

(a) $|M| = \emptyset$ then $x^M = \emptyset = t^M$, so $M \models \neg(t \doteq x)$ and the conclusion holds.

(b) $|M| \neq \emptyset$, we have the following subcases:

- $t^M = \emptyset$, then $M \models \neg(t \doteq x)$ and the conclusion holds. Similarly if $t'^M = \emptyset$.
- $|t^M| > 1$, then $M \models \neg(t \doteq x)$ and the conclusion holds. Similarly if $|t'^M| > 1$.
- $t^M = e \neq e' = t'^M$, then we must have either $M \models \neg(t' \doteq x)$ or $M \models \neg(t' \doteq x)$ since x^M can't be equal to both e and e'.

$\Uparrow$ Suppose that the conclusion holds. We have two cases:

(a) $|M| = \emptyset$ then $t^M = \emptyset = t'^M$, i.e., $M \models \neg(t \doteq t')$ and the premise holds.

(b) $|M| \neq \emptyset$, so if $M \models t \doteq x$, then for the conclusion to hold we must have $M \models \neg(t' \doteq x)$, i.e, $M \models \neg(t \doteq t')$. Similarly if $M \models t \doteq x$.

3. (X+)

$$\frac{\Gamma', t \doteq x, \Gamma''}{\Gamma', t \prec x, \Gamma'' \mid \Gamma', x \prec t, \Gamma'' \mid \Gamma', \neg\mathcal{E}, \Gamma''} \quad \text{where } x \in X \text{ and } t \neq x$$

$\Downarrow$ Suppose $M \models t_s \doteq x_s$ then:
$x^M = t^M = e \in |M|$, so $|M|$ is nonempty i.e $M \models \neg\mathcal{E}$ and $M \models t \prec x$ and $M \models x_s \prec t$.
$\Uparrow$ Suppose $M \models t \prec x$ and $M \models x \prec t$ and $M \models \neg\mathcal{E}$ then: $e = x^M \subseteq t^M \subseteq x^M = e$, i.e $e = x^M = t^M$, so $M \models t \doteq x$.

4. (X–)

$$\frac{\Gamma', \neg(t \doteq x), \Gamma''}{\Gamma', \neg(t \prec x), \neg(x \prec t), \mathcal{E}, \Gamma''} \quad \text{where } x \in X \text{ and } t \neq x$$

$\Downarrow$ Suppose that $M \models \neg(t \doteq x)$, two cases:

(a) $|M| = \emptyset$ then $M \models \mathcal{E}$.

(b) $|M| \neq \emptyset$, two possibilities:

i. There exists an element $e \in t^M$ such that $e \neq x^M$, then $M \models \neg(t \prec x)$, or

ii. $x^M = e'$ and $e' \notin t^M$, so $M \models \neg(x \prec t)$

$\Uparrow$ Three cases:

(a) $M \models \mathcal{E}$, then $M \models \neg(t \doteq x)$

(b) $M \models \neg(t \prec x)$, then there exists an element e such that $e \in t^M$ and $e \notin x^M$ so $M \models \neg(t \doteq x)$

(c) $M \models \neg(x \prec t)$, then there exists an element e such that $x^M = e$ and $e \notin t^M$ so $M \models \neg(t \doteq x)$

5. (XI+)

$$\frac{\Gamma', x_s \doteq t_s, \Gamma''}{\Gamma', t_s \prec x_s, \Gamma'' \mid \Gamma', x_s \prec t_s, \Gamma'' \mid \Gamma', \neg\mathcal{E}_s, \Gamma''} \quad \text{where } x \in X \text{ and } t \neq x$$

The proof of this rule is analogous to the proof of $(X+)$.

6. (XI–)

$$\frac{\Gamma', \neg(x_s \doteq t_s), \Gamma''}{\Gamma', \neg(t_s \prec x_s), \neg(x_s \prec t_s), \mathcal{E}_s, \Gamma''} \quad \text{where } x \in X \text{ and } t \neq x$$

—– The proof of this rule is analogous to the proof of (X–).

□

2.3 Completeness

We first show a lemma which gives the main part of the proof of completeness.

Lemma 2.10 *Given a set of indecomposable formulae, Γ_{ind}, that is closed under all expansion rules and that does not satisfy the axiomatic sequence condition, there exist a structure M_C, such that $M_C \not\models \gamma$, for every formula $\gamma \in \Gamma_{ind}$, i.e. M_C is a counter-model for Γ_{ind}.*

Proof. Given such a Γ_{ind}, we define the relation $\sim$ on the set X of variables by:[4]

$$x \sim y \iff \neg(y \prec x) \in \Gamma_{ind}$$

Closure under expansion rules implies that $\sim$ is symmetric, rule (XII), and transitive, rule (XIII).

The relation $\asymp$ is the reflexive closure of $\sim$, i.e.:

$$x \asymp y = \sim \ \cup \{(x,x) : x \in X\}$$

Again, closure under expansion rules implies that $\asymp$ is a congruence wrt. function applications: given $x \asymp y$ then if $\neg(x \prec f(z)) \in \Gamma_{ind}$, then also $\neg(y \prec f(z)) \in \Gamma_{ind}$, by rule (XIV), and if $\neg(z \prec f(x)) \in \Gamma_{ind}$, then also $\neg(z \prec f(y)) \in \Gamma_{ind}$, by rule (XV).

We now define the counter-model $M_C = \langle A_C, \alpha_C \rangle$ for Γ_{ind} as follows:

1. Carrier sets:
   (a) $|A_C|_s = \emptyset$ iff $\neg\mathcal{E}_s \in \Gamma_{ind}$
   (b) $|A_C|_s = X_s/_{\asymp}$ – otherwise
2. Operations: for $f : s_1 \times \ldots \times s_n \to s_{n+1}$, we define:
   (a) $f([\overline{x}])^{A_C} = \emptyset$, if $|A_C|_{s_i} = \emptyset$ for some $1 \leq i \leq n+1$
   (b) $f([\overline{x}])^{A_C} = \{[y] : \neg(y \prec f(\overline{x})) \in \Gamma_{ind}\}$
3. Assignment:
   (a) $\alpha(x) = \emptyset$ iff $\neg\mathcal{E}_s \in \Gamma_{ind}$
   (b) $\alpha(x) = [x]$ – otherwise

We prove that M_C is indeed a counter-model for Γ_{ind}, i.e., $M_C \not\models \gamma$, for every formula $\gamma \in \Gamma_{ind}$. We prove the statement for each type of indecomposable formula.

1. $\gamma = \mathcal{E}_s \in \Gamma_{ind}$, since Γ_{ind} is non-axiomatic it means that $\neg\mathcal{E}_s \notin \Gamma_{ind}$, hence: $|A_C|_s = X/_{\asymp} \neq \emptyset$, so $M_C \not\models \mathcal{E}_s$.
2. $\gamma = \neg\mathcal{E}_s \in \Gamma_{ind}$, so $|A_C|_s = \emptyset$, and we have that $M_C \not\models \neg\mathcal{E}_s$
3. $\gamma = x_s \prec y_s \in \Gamma_{ind}$, then $\neg\mathcal{E}_s \notin \Gamma_{ind}$ (otherwise Γ_{ind} would be axiomatic) and so $|A_C|_s = X_s/_{\asymp}$ and $\alpha(x) = [x] \neq \emptyset \neq [y] = \alpha(y)$. Then $M_C \models x \prec y \iff [x] \subseteq [y]$, but this holds only if $[x] = [y]$, i.e., only if $\neg(x_s \prec y_s) \in \Gamma_{ind}$, which is not the case since Γ_{ind} is not axiomatic.
4. $\gamma = \neg(x_s \prec y_s) \in \Gamma_{ind}$, then we have the following subcases:
   (a) $|A_C|_s = \emptyset$ then $\alpha(x) = \emptyset = \alpha(y)$, so $M_C \not\models \neg(x_s \prec y_s)$
   (b) $|A_C|_s = X/_{\asymp}$, then $\alpha(x) = [x] \neq \emptyset \neq [y] = \alpha(y)$. Then $M_C \models x \prec y \iff [x] \subseteq [y]$, but this holds since $\neg(x_s \prec y_s) \in \Gamma_{ind}$, i.e. $M_C \not\models \neg(x_s \prec y_s)$.

[4] We assume, in general, that the set X of all variables is countable. If it is not, we choose here only the variables which occur in some formula in Γ_{ind}.

5. $\gamma = x_s \prec f_s(x_1, \ldots, x_n) \in \Gamma_{ind}$, since Γ_{ind} is non-axiomatic we have $\neg\mathcal{E}_s \notin \Gamma_{ind}$, and $|A_C|_s = X/_{\asymp}$. We have the following subcases:
   (a) $\alpha_C(x_i) = \emptyset$ for some $1 \le i \le n$, then $|A_C|_{s_i} = \emptyset$ by definition of assignment, (and also $\neg\mathcal{E}_{s_i} \in \Gamma_{ind}$ by definition of M_C 1a). But then $f^{M_C}(x_1, \ldots, x_n) = \emptyset$ by 2a, while $\alpha_C(x) = [x] \ne \emptyset$, so $M_C \not\models x_s \prec f_s(x_1, \ldots, x_n)$.
   (b) For all $1 \le i \le n : |A_C|_{s_i} \ne \emptyset$. Then $[x] \in f([x_1], \ldots, [x_n])$ iff $\neg(x \prec f(x_1, \ldots, x_n)) \in \Gamma_{ind}$, but this is a contradiction since Γ_{ind} is non-axiomatic, so $M_C \not\models x_s \prec f_s(x_1, \ldots, x_n)$.
6. $\gamma = \neg(x_s \prec f_s(x_1, \ldots, x_n)) \in \Gamma_{ind}$, we have the following subcases:
   (a) $|A_C|_s = \emptyset$, then $f^{A_C}(x_1, \ldots, x_n) = \emptyset$, so $M_C \not\models \neg(x_s \prec f_s(x_1, \ldots, x_n))$.
   (b) $|A_C|_s = X_s/_{\asymp}$, we have two subcases:
      i. $\alpha_C(x_i) = \emptyset$ for some $1 \le i \le n$, then we have for the sort s_i of x_i that $\neg\mathcal{E}_{s_i} \in \Gamma_{ind}$. Since $\neg(x_s \prec f_s(x_1, \ldots, x_n)) \in \Gamma_{ind}$ and Γ_{ind} is closed under expansion rules we get by the expansion rule (XVI) that $\neg\mathcal{E}_s \in \Gamma_{ind}$, this is a contradiction since $\neg\mathcal{E}_s \in \Gamma_{ind}$ implies that $|A_C|_s = \emptyset$.
      ii. All the carriers of the sorts of the variables x_i are nonempty. So we have $[x] \in f([x_1], \ldots, [x_n])$ iff $\neg(x \prec f(x_1, \ldots, x_n)) \in \Gamma_{ind}$, and since the latter holds we get that $M_C \not\models \neg(x \prec f(x_1, \ldots, x_n))$.

$\square$

Using this lemma, we obtain the main completness theorem.

Theorem 2.11 *The R-S system is complete: if* $\models \Gamma$ *(i.e.,* $\forall M = \langle A, \alpha \rangle : M \models \Gamma$*), then* $\vdash \Gamma$.

PROOF. We show that if $\not\vdash \Gamma$, then $\not\models \Gamma$, i.e., there exists a structure M_C with $M_C \not\models \Gamma$. Let $DT(\Gamma)$ be the unique decomposition tree for Γ as defined in def. 2.6. There are two situations when $DT(\Gamma)$ is not a proof:

I. Some leaves are labelled by non-axiomatic sequences – then such leaves have labels containing only indecomposable formulae, or

II. The tree is infinite, which implies (by the König lemma) that there exists an infinite branch.

In either case we can find a set Γ_{ind} of indecomposable formulae closed under all expansion rules which is valid if Γ is valid. Thus, a counter-model M_C for Γ_{ind}, which exists by lemma 2.10, is also a counter-model for Γ.

I. The non-axiomatic sequence labeling one of the leaves can be taken as Γ_{ind} – by definition of $DT(\Gamma)$, Γ_{ind} is closed under all expansion rules. Since $M \models \Gamma$ implies $M \models \Gamma_{ind}$, lemma 2.10, giving a counter-model $M_C \not\models \Gamma_{ind}$, implies that $M_C \not\models \Gamma$.

II. Select an infinite branch B from $DT(\Gamma)$. If an indecomposable formula appears at some vertex of B, then it appears also at all subsequent vertices. Let Γ_{ind} be the union (infinite set) of all indecomposable formulae appearing in the labels of the vertices on the branch B. Γ_{ind} does not satisfy the axiomatic sequence condition, for if it does, then there exists a vertex at which this axiom occurs and which would terminate B. Also, by lemma 2.8, Γ_{ind} is closed under all expansion rules. Thus Γ_{ind} satisfies the conditions of lemma 2.10, so let M_C be the counter-model as it was defined in the proof of this lemma.[5] We show that M_C is a counter-model for all the formulae occurring in the labels of the vertices of B, and since the root vertex of B is the root of $DT(\Gamma)$, i.e., is labelled with Γ, we have that M_C is a counter-model for Γ. The proof goes by induction on the rank of a formula γ, ord(γ), which we define so that the applications of rules never increase the rank of the formulae in the sequence and, eventually, decrease it.

- $\mathrm{ord}(\gamma) = 0$, if γ is indecomposable;
- otherwise:
  - $\mathrm{ord}(x \prec t) = \mathrm{ord}(\neg(x \prec t)) = 1$ (where $t \notin X$, $x \in X$)
  - $\mathrm{ord}(t \prec t') = \mathrm{ord}(\neg(t \prec t')) = 2$ (where $t \notin X$)
  - $\mathrm{ord}(x \doteq t) = \mathrm{ord}(t \doteq x)) = \mathrm{ord}(\neg(x \doteq t)) = \mathrm{ord}(\neg(t \doteq x)) = 2$ (where t may be a variable, even x)

[5] Here, as well as in the point I., the carrier of the counter-model M_C can be constructed only from the variables occuring in Γ_{ind}.

– $\mathrm{ord}(t \doteq t') = \mathrm{ord}(\neg(t \doteq t')) = 3$ (where both $t \notin X$ and $t' \notin X$)
– $\mathrm{ord}(\gamma' \vee \gamma'') = \mathrm{ord}(\gamma' \wedge \gamma'') = \max(\mathrm{ord}(\gamma'), \mathrm{ord}(\gamma'')) + 1$.
– $\mathrm{ord}(\neg\gamma') = \mathrm{ord}(\gamma') + 1$, if γ' is not any of the above cases.

Now, if $M_C \models \Gamma$, then the set Γ_{sat} of all formulae γ', appearing in one of the vertices of B and such that $M_C \models \gamma'$, is nonempty, since $\Gamma \cap \Gamma_{sat} \neq \emptyset$. Let $\gamma_i \in \Gamma_{sat}$ be such that $\mathrm{ord}(\gamma_i) \leq \mathrm{ord}(\gamma')$, for every $\gamma' \in \Gamma_{sat}$. We show, by induction on the rank of γ_i, that it must be indecomposable.

Suppose that γ_i is decomposable. By point 2. of lemma 2.8, there exists a vertex $\Pi_i \in B$ such that $\Pi_i = \Gamma', \gamma_i, \Gamma''$, where Γ' is indecomposable (possibly empty) and closed under all expansion rules, and Π_{i+1} is the vertex following Π_i in B, with the label $\Gamma', \gamma_{i+1}, \Gamma'''$ obtained by the correct application of a decomposition rule. Considering the possible cases for γ_i, we show that there exists a $\gamma' \in \Gamma_{sat}$ with $\mathrm{ord}(\gamma') < \mathrm{ord}(\gamma_i)$, which contradicts the assumption about γ_i:

1. $\phi \vee \mu, \neg(\phi \vee \mu), \phi \wedge \mu, \neg(\phi \wedge \mu), \neg\neg\phi$. If γ_i has one of these forms, then $\mathrm{ord}(\gamma_{i+1}) < \mathrm{ord}(\gamma_i)$ and $M_C \models \gamma_{i+1}$, which contradicts the definition of γ_i.
2. $\neg(t \doteq t')$.
   (a) If neither $t, t' \notin X$, then by rule (IX–), $\Pi_{i+1} = \Gamma', \neg(t \doteq x), \neg(t' \doteq x), \Gamma''$. We then have that $M_C \models \gamma'$, where γ' is either $\neg(t \doteq x)$ or $\neg(t' \doteq x)$. In either case $\mathrm{ord}(\gamma') = 2 < 3 = \mathrm{ord}(\gamma_i)$ which contradicts the definition of γ_i.
   (b) If $t = x \in X$ or $t' = x \in X$ then, by rule (X–) or (XI–), $\Pi_{i+1} = \Gamma', \neg(t \prec x), \neg(x \prec t), \Gamma''$, and if $M_C \models \gamma_i$ then $M_C \models \gamma'$ where γ' is either $\neg(t \prec x)$ or $\neg(x \prec t)$. Hence $\mathrm{ord}(\gamma') \leq 1 < 2 = \mathrm{ord}(\gamma_i)$, which contradicts the definition of γ_i. ($\mathrm{ord}(\gamma') < 1$ when both $t, t' \in X$.)
3. $t \doteq t'$.
   (a) If $t \notin X$ and $t' \notin X$ then, by rule (IX+), γ_{i+1} is $t \doteq x$ (if B follows $DT(\Gamma)$ along the left conclusion), or $t' \doteq x$ (if B proceeds along the right conclusion – both cases are entirely analogous). If $M_C \models \gamma_{i+1}$ then we are done, since in either case $\mathrm{ord}(\gamma_{i+1}) = 2 < 3 = \mathrm{ord}(\gamma_i)$. However, it may happen that $M_C \not\models \gamma_{i+1}$ (because of a wrong choice of the variable x). Then Π_{i+1}, as well as all other vertices along B, inherit $t \doteq t'$ as $\Gamma', t \doteq x, \Gamma'', t \doteq t'$. [6] At some point, the trailing $t \doteq t'$ will be processed anew according to rule (IX+), introducing a new variable: $t \doteq y$ (or in $t' \doteq y$). Eventually, since $M_C \models t \doteq t'$, we will get the appropriate variable, say z, such that $M_C \models t \doteq z$ (or $M_C \models t' \doteq z$).[7] Satisfaction of this formula contradicts the assumption about $\gamma_i = (t \doteq t')$, since $\mathrm{ord}(t \doteq z) = 2 < 3 = \mathrm{ord}(t \doteq t')$.
   (b) If $t \in X$ or $t' \in X$ then γ_{i+1} has the form $t \prec x$ or $x \prec t$ (or t' instead t) or $\neg\mathcal{E}$ (by rule (X+) or (XI+)) and hence $\mathrm{ord}(\gamma_{i+1}) \leq 1 < 2 = \mathrm{ord}(\gamma_i)$, which contradicts the definition of γ_i. ($\mathrm{ord}(\gamma_{i+1}) = 0$ if $\gamma_{i+1} = \neg\mathcal{E}$.)
4. $\neg(t \prec t')$.
   (a) If $t \notin X$ then $\Pi_{i+1} = \Gamma', \gamma_{i+1}, \Gamma'', \neg(t \prec t')$. The argument is entirely analogous to that in point 3a. B follows $DT(\Gamma)$ either along the left conclusion of the rule (VII–) with $\gamma_{i+1} = x \prec t$, or along the right one with $\gamma_{i+1} = \neg(x \prec t')$. In either case $\mathrm{ord}(\gamma_{i+1}) = 1 < 2 = \mathrm{ord}(\gamma_i)$, so if $M_C \models \gamma_{i+1}$, we are done.[8] Eventually, the trailing $\neg(t \prec t')$, inherited in Π_{i+1} and all following vertices in B, will be processed again and again along B according to the rule (VII–) providing, eventually, a witness variable, say y, such that $M_C \models y \prec t$ – if B happens to proceed along a left conclusion, or $M_C \models \neg(y \prec t')$ – if B proceeds along the right conclusion. In either case, the satisfied formula has lower rank than γ_i which contradicts its definition.
   (b) If $t \in X$ then, as γ_{i+1} is decomposable, $t' = f(\ldots, t'', \ldots)$ with $t'' \notin X$. By rule (VIII–), $\Pi_{i+1} = \Gamma', \neg(y \prec t''), \neg(t \prec f(\ldots, y, \ldots)), \Gamma''$, and the assumption $M_C \models \neg(t \prec t')$ implies $M_C \models \neg(y \prec t''), \neg(t \prec f(\ldots, y, \ldots))$. Before any branching of $DT(\Gamma)$, i.e.,

[6] If $M_C \models \gamma''$ for some formula $\gamma'' \in \Gamma''$ with $\mathrm{ord}(\gamma'') = 3 = \mathrm{ord}(t \doteq t')$, then γ'' is either of the form $s \doteq s'$ or $\neg(s \doteq s')$. The latter case was covered in point 2 and the former is the same as the current case. (It may also be of the form $\phi \wedge \psi, \phi \vee \psi$ or $\neg\phi$ but all these cases have been covered in point 1.)

[7] Since the carrier $|A_C|$ of M_C is constructed as a quotient of the variable set X, the assignment α_C is surjective.

[8] If $M_C \models \gamma''$ for some $\gamma'' \in \Gamma''$ with $\mathrm{ord}(\gamma'') = 2 = \mathrm{ord}(\gamma_i)$, then γ'' has either the form $s \prec s'$ or $\neg(s \prec s')$. The former case is treated in point 5 below, while the latter is the present case.

still along the branch B, the rule (VIII–) (and possibly some expansion rules) will be applied to both these formulae until we get only indecomposable formulae. Since $M_C \models \gamma_i = \neg(t \prec t')$, we must have $M_C \models \gamma'$ for one of these indecomposable formulae, which contradicts the definition of γ_i.

5. $t \prec t'$.
   (a) If $t \notin X$ then, by rule (VII+), $\Pi_{i+1} = \Gamma', \neg(x \prec t'), x \prec t', \Gamma''$. Then $M_C \models t \prec t'$ implies $M_C \models \neg(x \prec t'), x \prec t'$. But each of these two formulae has rank lower than γ_i, which contradicts its definition.
   (b) If $t = x \in X$ then $t' = f(\ldots, t'', \ldots)$ and B follows $DT(\Gamma)$ either along the left or along the right conclusion of rule (VIII+), with $\gamma_i = x \prec f(\ldots, t'', \ldots)$ repeated at the end of the sequence. The argument is analogous to those in points 3a and 4a. γ_{i+1} has the form $y \prec t''$ (if B follows the left branch), or $x \prec f(\ldots, y, \ldots)$ (if B follows the right branch), and in either case the rule (VIII+) is applied until γ_{i+1} becomes indecomposable formula. If M_C satisfies it, we get a contradiction with the definition of γ_i. Otherwise, the trailing repetition of γ_i has to be processed again and again along B, and the above argument leads to an indecomposable formula which has to be satisfied, thus yielding a contraiction with the definition of γ_i.

Thus we have shown that a formula γ_i which appears on an infinite branch B of $DT(\Gamma)$, which is satisfied, $M_C \models \gamma_i$, and which is of lowest rank among the formulae satisfying these two conditions, has to have rank 0, i.e., must be indecomposable, which means that $\gamma_i \in \Gamma_{ind}$. But then $M_C \not\models \gamma_i$, by the construction of M_C in lemma 2.10, which contradicts the assumption that $M_C \models \gamma_i$. □

Corollary 2.12 *A sequence Γ has a proof in the R-S system iff $DT(\Gamma)$ is a proof.*

Proof. The 'if' part is trivial and the 'only if' part follows from the proof of the above theorem. If $DT(\Gamma)$ is not a proof, then we have a counter-model for Γ. Since R-S is sound, we conclude that Γ is not provable. □

3 Specifications and system R-S*

The system R-S can be used to derive only tautologies – valid sequences. But we are really interested in proving logical consequences of specifications. A specification $\mathcal{SP}$ is pair (Σ, Ψ), where Σ is a signature and Ψ is a set of axioms which we will write as sequents of atomic formulae. (For the purposes of this paper, we can safely ignore the signature assuming that all expressions are well-formed, and identify a specification with the set of its axioms.) We are interested in proving sequents which follow logically from such specifications. In this section we extend the R-S logic to fulfill this function. (In the following section we will return to the sequent form and transform the R-S* logic into a sound and complete Gentzen system.)

3.1 Specifications

Specifications are sets of sequents from which one may derive other sequents.

Definition 3.1 *A Σ sequent is a pair (Γ, Δ) of finite sets of formulae from $\mathcal{F}_{\Sigma,X}$, written $\Gamma \to \Delta$.*

The notation $\Gamma \to \Delta$ is implicitly assumed to mean the same as $\gamma_1, \ldots, \gamma_n \to \delta_1, \ldots, \delta_m$. As a matter of fact, following earlier works, e.g. [3, 13, 7], our specifications involve only sequents of atomic formulae (i.e., each γ_i, δ_j is either equality or inclusion), but we may occasionally need this more general definition. Keep also in mind that all formulae in a sequent are quantifier free.

Definition 3.2 *A Σ sequent $\Gamma \to \Delta$ is valid iff for every Σ-structure $M = \langle A, \alpha \rangle$ such that $M \models \gamma$, for all $\gamma \in \Gamma$, there exists a $\delta \in \Delta$ such that $M \models \delta$.*

Lemma 3.3 *A sequent $\Gamma \to \Delta$ is valid iff the sequence $\neg\Gamma, \Delta$ is valid.*

The latter notation stands for the sequence $\neg\gamma_1, \ldots, \neg\gamma_n, \delta_1, \ldots, \delta_m$, where $\gamma_1, \ldots, \gamma_n \to \delta_1, \ldots, \delta_m$ is the respective sequent.

Definition 3.4 *The function* tr *translates sequents to (quantifier free) formulae in* $\mathcal{F}_{\Sigma,X}$*:*

- $\mathsf{tr}(\gamma_1, \ldots, \gamma_n \to \delta_1, \ldots, \delta_m) \equiv \neg\gamma_1 \vee \ldots \vee \neg\gamma_n \vee \delta_1 \vee \ldots \vee \delta_m$.
- *for* $\Psi = \{\psi_1, ..., \psi_n\} : \mathsf{tr}(\Psi) = \{\mathsf{tr}(\psi_1), ..., \mathsf{tr}(\psi_n)\}$

With the above notation, lemma 3.3 can be stated as: for any structure $M = \langle A, \alpha\rangle$ and sequent $\psi : M \models \psi \iff M \models \mathsf{tr}(\psi)$

The models for a specification are no longer structures with an assignment, but algebras satisfying the axioms for all possible assignments:

Definition 3.5 *Given a specification* $\mathcal{SP} = (\Sigma, \Psi)$*, a* Σ*-algebra* A*, and a sequent (formula, sequence)* ψ*, we define the satisfaction relation* $\models_*$*:*

1. $A \models_* \psi \iff \forall\alpha.\langle A, \alpha\rangle \models \mathsf{tr}(\psi)$
2. $A \models_* \Psi \iff \forall\psi_i \in \Psi.A \models_* \mathsf{tr}(\psi_i)$
3. $\Psi \models_* \psi \iff \forall A.(A \models_* \Psi \Rightarrow A \models_* \mathsf{tr}(\psi))$.

Notice that in the case of tautologies the two notions of satisfiability coincide, i.e., $\models \psi \iff \models_* \psi$. The above definition may be applied also when ψ's are arbitrary formulae, in which case we simply drop the applications of $\mathsf{tr}(\psi)$. This convention will be applied below – ψ stands, in general, for arbitrary formula, while the notation $\mathsf{tr}(\psi)$ indicates that ψ is a sequent.

3.2 System R-S*

We introduce the syntax for indicating axiomatic sequents, define their semantics (reflecting the intended relation $\models_*$), and extend the system R-S with a new rule to handle such sequents.

Definition 3.6 *An axiom* ψ *is written* $![\psi]$

The procedure for extending the R-S system is quite standard – in order to prove a sequent ψ from a specification (set of sequents) $\Psi = \{\psi_1, \ldots, \psi_n\}$, we perform a translation, tr, of ψ and all the sequents from Ψ into formulae, form a sequence corresponding to $(\bigwedge_{\psi_i \in \Psi} ![\mathsf{tr}(\psi_i)]) \to \mathsf{tr}(\psi)$, and try to prove it in the system R-S augmented with the appropriate rule for treating axioms on the left of '→'. The standard notion of satisfaction of such a formula is equivalent to the satisfaction of a sequence

$$\neg![\mathsf{tr}(\psi_1)], \ldots, \neg![\mathsf{tr}(\psi_n)], \mathsf{tr}(\psi) \tag{1}$$

The details concerning the corresponding Gentzen system will be given in Section 4. For the time being we merely observe that in order to reason about specifications we have to extend the R-S proof system by a new rule to handle axiomatic formulae, i.e., the formulae with the form $\neg![\phi]$. (Notice that in (1) we do not nest axiomatic formulae, and they always occur under the negation $\neg$. Since specifications will only involve sequents over atomic formulae, we do not need the full power of universal and/or existential quantifiers.) Therefore we introduce $![\_]$, resp. $\neg![\_]$ as new logical connectives which, however, are used only at the outermost level of formulae.

Definition 3.7 *For a structure* $M = \langle A, \alpha\rangle$ *and a formula* ψ*, we define:*

- $M \models ![\psi] \iff A \models_* \psi$ *(i.e., iff* $\forall\alpha'.\langle A, \alpha'\rangle \models \psi$*). Consequently:*
- $M \models \neg![\psi] \iff M \not\models ![\psi] \iff A \not\models_* \psi$ *(i.e., iff* $\exists\alpha'.\langle A, \alpha'\rangle \models \neg\psi$*).*

As usual, α' matter only in so far as it differs from α on the variables occuring in ψ.

Notice that we quantify over *assignments* α' – according to definition 1.4 such an assignment may exist even if the carrier A is empty, in which case all variables are assigned $\emptyset$. $![\_]$ does play the role of the universal closure but over assignments and not only elements of the carrier. Similarly $\neg![\_]$ corresponds to existential closure over assignments.

$M[\alpha'/\alpha]$ denotes the structure M with α replaced by α'. Similarly, for a formula ϕ, we write $\phi[\overline{y}/\overline{x}]$ for ϕ with all occurrences of the variables from the sequence $\overline{x}$ replaced by the respective variables from the sequence $\overline{y}$.

The R-S* system is obtained by augmenting the R-S system with the following rule:

$$\text{(AX)}\ \frac{\Gamma', \neg![\gamma], \Gamma''}{\Gamma', \neg\gamma[\overline{y}/\overline{x}], \Gamma'', *}\quad \text{where } \overline{x} \text{ are all variables in } \gamma\text{, and variables } \overline{y} \text{ are arbitrary}$$

Lemma 3.8 *The R-S* system is sound:*

Proof. The R-S system is sound so it remains to prove soundness of the new rule. We let M be an arbitrary structure $M = \langle A, \alpha \rangle$

$\Downarrow$ If $M \models \neg ! [\gamma]$, then M obviously satisfies the conclusion of the rule since this formula is repeated there.

$\Uparrow$ If $M \models \neg\gamma[\overline{y}/\overline{x}]$, we let $\alpha'(\overline{x}) = \alpha(\overline{y}) \in |A| \uplus \{\emptyset\}$, so $M[\alpha'/\alpha] \models \neg\gamma$. By the definition 3.7 we have that: $M \models \neg ! [\gamma]$. Every other formula from the conclusion appears also in the premise, so all these cases are trivial.

□

Remark 3.9 *As one could expect, the definition of satisfaction 3.7 makes $\neg ! [\gamma]$ equivalent to its standard counterpart $\exists\alpha : A \not\models_* \gamma$. The non-standard aspect is that such quantification over assignments does not coincide with the quantification over elements of the carrier, since in case of empty carrier we still admit the assignment $\alpha(x) = \emptyset$.*

Consider the following special cases, with $\Gamma' = \emptyset = \Gamma''$, of the application of rule (AX):

1. *If γ is $\neg(x_s \doteq x_s)$, we get:*

$$(AX)\ \ \frac{\neg ! [\neg(x_s \doteq x_s)]}{\neg\neg(y_s \doteq y_s), *}\ x_s \in X$$

Applying (IV–) to the conclusion, we obtain $y_s \doteq y_s$, i.e., $\neg\mathcal{E}_s$. Thus the formulae $\neg ! [\neg(x_s \doteq x_s)]$ and $\neg\neg(x_s \doteq x_s) \equiv \neg\mathcal{E}_s$, are really equivalent, i.e. $\exists\alpha : x \doteq x$ is equivalent to $\exists x : x \doteq x$. (If the carrier is empty, there is not only no element but also no assignment making $x \doteq x$, since $\emptyset$ does not satisfy this equality.)

2. *If γ is $x_s \doteq x_s$, we get:*

$$(AX)\ \ \frac{\neg ! [x_s \doteq x_s]}{\mathcal{E}_s, *}\ x_s \in X$$

where $\mathcal{E}_s$ in the conclusion corresponds to $\neg(y_s \doteq y_s)$, for some variable y_s substituted for x_s.

Thus $\neg ! [x_s \doteq x_s]$ and $\neg(x_s \doteq x_s) \equiv \mathcal{E}_s$, are equivalent, and correspond to $\exists\alpha : \neg(x \doteq x)$ which is satisfied only by the structures with empty carrier. Note, however, that this is not equivalent to $\exists x : \neg(x \doteq x)$ – this last formula is actually a contradiction.

In earlier logics of ours, e.g. [12, 13], we did not admit empty carrier and then $x \doteq x$ was axiomatic. The generalization with this respect amounts to having made this formula valid if and only if carrier is non-empty. The significant difference with respect to [2] is that our treatement of (non-)empty carrier is essentially quantifier-free – it amounts merely to the treatement of the formulae $x \doteq x$ (resp. $\neg(x \doteq x)$) which is carried over to the respective axioms as shown in the remark above. In [2], this required formulae of the form $\exists x.x \prec x$ (resp. $\neg\exists x.x \prec x$).

Lemma 3.10 *The R-S* proof system is complete: for any sequence Γ (of formulae from $\mathcal{F}_{\Sigma,X}$ or, possibly, of the form (1)), if $\models \Gamma$ then $\vdash \Gamma$.*

Proof. The only difference from the R-S proof system is the presence of the new kind of formulae and the new rule (AX). Note that the R-S* system has the same axiomatic sequences and the same indecomposable formulae as the R-S system. The proof is therefore the same as before, based on the counter-model M_C from lemma 2.10. We only have to consider a new possible case of a formula γ_i which, occuring on the selected infinite branch B (from which we constructed the counter-model M_C), has the lowest rank such that $M_C \models \gamma_i$. (Lemmata 2.7 and 2.8 remain trivially true for the R-S* system. To obtain unique decomposition tree, we would have to extend the well-ordering of variables from footnote 3 to finite sequences of variables since rule ((AX)) performs uniform substitution for *all* variables in the processed formula.) We define the rank of the formula $\neg ! [\gamma]$ by:

- $\mathrm{ord}(\neg ! [\gamma]) = \mathrm{ord}(\neg\gamma) + 1$

As in the proof of theorem 2.11, let $\Pi_i \in B$ be such that $\Pi_i = \Gamma', \neg ! [\gamma], \Gamma''$, where Γ' is indecomposable (possibly empty), and Π_{i+1} be the vertex following Π_i in B with the label $\Gamma', \gamma_{i+1}, \Gamma'''$.

6. If B follows the conclusion, then, eventually, there must be a vertex $\Pi_j \in B$, $j > i$, including the formula $\neg\gamma[\overline{y}/\overline{x}]$ such that $M_C \models \neg\gamma[\overline{y}/\overline{x}]$, since $M_C \models \gamma_i$ (and since the carrier of M_C is obtained as a quotient of the variable set X, with $\alpha_C(y) = [y]$). Whether this happens already for $j = i+1$ or later does not matter since in either case we have: $\mathrm{ord}(\neg\gamma[\overline{y}/\overline{x}]) = \mathrm{ord}(\neg\gamma) < \mathrm{ord}(\neg\gamma) + 1 = \mathrm{ord}(\gamma_i)$.

□

We introduce the following notational abbreviations:

Definition 3.11 *Given a set of formulae $\Psi = \{\psi_1, \ldots, \psi_n\} \subseteq \mathcal{F}_{\Sigma,X}$ and a $\psi \in \mathcal{F}_{\Sigma,X}$, we write*

- $\Psi \vdash \psi \iff \vdash \neg![\psi_1], \ldots, \neg![\psi_n], \psi$
- $\Psi \models \psi \iff \models \neg![\psi_1], \ldots, \neg![\psi_n], \psi$

The above lemmata 3.8, 3.10 give us:

Theorem 3.12 *For any formula ϕ and set of formulae $\Phi = \{\phi_1, \ldots, \phi_n\}$: $\Phi \vdash \phi \iff \Phi \models \phi$.*

Corollary 3.13 *For any specification Ψ and sequent ψ : $\Psi \vdash \psi \iff \Psi \models_* \psi$.*

Proof. By the above theorem 3.12, we only have to show, for any specification Ψ and sequent $\psi : \Psi \models \psi \iff \Psi \models_* \psi$.

We have:
$$\begin{aligned}
\Psi \models \psi &\iff \mathsf{tr}(\Psi) \models \mathsf{tr}(\psi) \\
&\overset{3.11}{\iff} \models \neg![\mathsf{tr}(\psi_1)] \vee \ldots \vee \neg![\mathsf{tr}(\psi_1)] \vee \mathsf{tr}(\psi) \\
&\iff \models_* \neg![\mathsf{tr}(\psi_1)] \vee \ldots \vee \neg![\mathsf{tr}(\psi_1)] \vee \mathsf{tr}(\psi) \\
&\iff \forall A.\ A \models_* \bigvee_{\psi_i \in \Psi} \neg![\mathsf{tr}(\psi_i)] \vee \mathsf{tr}(\psi)
\end{aligned}$$

And:
$$\begin{aligned}
\Psi \models_* \psi &\overset{3.5}{\iff} \forall A.\ (A \models_* \Psi \Rightarrow A \models_* \mathsf{tr}(\psi)) \\
&\iff \forall A.\ (A \models_* \bigwedge_{\psi_i \in \Psi} \mathsf{tr}(\psi_i) \Rightarrow A \models_* \mathsf{tr}(\psi))
\end{aligned}$$

We thus have to show the following equivalence:

$$\forall A.\Big(A \models_* \bigwedge_{\psi_i \in \Psi} \mathsf{tr}(\psi_i) \ \Rightarrow\ A \models_* \mathsf{tr}(\psi)\Big) \quad\iff\quad \forall A.\Big(A \ \models_* \bigvee_{\psi_i \in \Psi} \neg![\mathsf{tr}(\psi_i)] \vee \mathsf{tr}(\psi)\Big)$$

$\Leftarrow$) Assume the RHS, and let A be arbitrary:

$$\begin{aligned}
A \models_* \bigwedge \mathsf{tr}(\psi_i) &\overset{3.7}{\iff} \forall\alpha.(\ \langle A,\alpha\rangle \models \bigwedge ![\mathsf{tr}(\psi_i)]\ ) \\
&\iff \forall\alpha.(\ \langle A,\alpha\rangle \not\models \bigvee \neg![\mathsf{tr}(\psi_i)]\ ) \\
&\overset{RHS}{\Longrightarrow} \forall\alpha.(\ \langle A,\alpha\rangle \models \mathsf{tr}(\psi)\ ) \\
&\iff A \models_* \mathsf{tr}(\psi)
\end{aligned}$$

$\Rightarrow$) Assume LHS and choose arbitrary A and α. If $\langle A,\alpha\rangle \models \bigvee \neg![\mathsf{tr}(\psi_i)]$, then RHS is satisfied.

So assume the opposite, i.e.:

$$\begin{aligned}
\langle A,\alpha\rangle \not\models \bigvee \neg![\mathsf{tr}(\psi_i)] &\iff \langle A,\alpha\rangle \models \bigwedge ![\mathsf{tr}(\psi_i)] \\
&\overset{3.7}{\iff} A \models_* \bigwedge \mathsf{tr}(\psi_i) \\
&\overset{LHS}{\Longrightarrow} A \models_* \mathsf{tr}(\psi) \\
&\Longrightarrow \langle A,\alpha\rangle \models \mathsf{tr}(\psi)
\end{aligned}$$

□

We have thus obtained the sound and complete system for proving consequences of specifications. As remarked, the system R-S*, with the unique proof strategy described in Section 2.1, is well suited for implementation. It is, however, less convenient for doing proofs by hand. In the following section we make the last step and design a Gentzen system which provides simpler means for performing proofs by hand – it works directly with sequents and does not require any translation of sequents into formulae.

4 Gentzen calculus

We will first describe a trivial translation of the R-S* system into a Gentzen system, GS', and then simplify it to the system GS", which we show to be equivalent to GS'. The final Gentzen system GS, given in Section 4.1, will be obtained by some further simplifications of GS".

Definition 4.1 *We say that a formula γ (in $\mathcal{F}_{\Sigma,X}$, or of the form $\neg\forall x\ldots$) is negative if it has the form $\neg\gamma'$, and non-negative else. For any sequence Γ we define:*

- $\Gamma^+ = \{\gamma \in \mathcal{F}_\Sigma : \gamma$ *is non-negative and* $\gamma \in \Gamma\}$
- $\Gamma^- = \{\gamma \in \mathcal{F}_\Sigma : \neg\gamma \in \Gamma\}$

We can now rephrase lemma 3.3 in the following way:

Corollary 4.2 *A sequence of formulae Γ is valid iff the sequent $\Gamma^- \to \Gamma^+$ is valid.*

The Genzen system we will design has three fundamental differences from the R-S system:

- The rules in Gentzen system are applied "bottom up" (hence the inversion of the R-S rules).
- The rules in Gentze system are not invertible – they are sound "top down", i.e., if the premise (above the stroke) is valid then so is the conclusion.
- Sequents are pairs of *sets* of formulae, where sequence ordering is ignored.

We use the corollary 4.2 on the different types of R-S rules and get the following lemma. Π stands for the active (sub)sequence of an R-S rule and Λ for the resulting (sub)sequence. Γ' and Γ'' in the R-S rules are arbitrary, so they are replaced in Gentzen rules by arbitrary Γ's and Δ's.

Lemma 4.3 *For any sound R-S rule (in the left column), the corresponding sequent (in the right column) is sound:*

R-s rule	*Gentzen rule*
$\dfrac{\Gamma', \Pi, \Gamma''}{\Gamma', \Lambda, \Gamma''}$	$\dfrac{\Gamma, \Lambda^- \to \Delta, \Lambda^+}{\Gamma, \Pi^- \to \Gamma, \Pi^+}$
$\dfrac{\Gamma', \Pi, \Gamma''}{\Gamma', \Lambda_1, \Gamma'' \mid \Gamma', \Lambda_2, \Gamma''}$	$\dfrac{\Gamma, \Lambda_1^- \to \Delta, \Lambda_1^+ \mid \Gamma, \Lambda_2^- \to \Delta, \Lambda_2^+}{\Gamma, \Pi^- \to \Gamma, \Pi^+}$
$\dfrac{\Gamma', \Pi, \Gamma''}{\Gamma', \Lambda_1, \Gamma'' \mid \Gamma', \Lambda_2, \Gamma'' \mid \Gamma', \Lambda_3, \Gamma''}$	$\dfrac{\Gamma, \Lambda_1^- \to \Delta, \Lambda_1^+ \mid \Gamma, \Lambda_2^- \to \Delta, \Lambda_2^+ \mid \Gamma, \Lambda_3^- \to \Delta, \Lambda_3^+}{\Gamma, \Pi^- \to \Gamma, \Pi^+}$

Applying the translation schema from lemma 4.3 to all the rules and axioms of the R-S system yields a Gentzen system GS', to which we add one rule: (IV+) $\dfrac{\Gamma, \gamma \to \Delta}{\Gamma \to \Delta, \neg\gamma}$.

Theorem 4.4 *The system GS' is sound and complete, i.e., for any sequent $\psi : \vdash_{GS'} \psi \iff \models \psi$.*

Proof. The system is sound by lemma 4.3 – soundness (and invertibility) of the rule (IV+) is obvious and so, by completeness of the R-S system, this rule is admissible there.

If a sequent $\Gamma \to \Delta$ is valid, then the corresponding R-S sequence $\neg\Gamma, \Delta$ is also valid. Since the R-S system is complete it means that $\neg\Gamma, \Delta$ has a proof in the R-S system, i.e. a finite decomposition tree T with leaves labeled by axiomatic sequences. We can then construct a Gentzen proof of the sequent $\Gamma \to \Delta$ in GS', by mimicking the structure of this decomposition tree w

The construction starts at the leaves of the deduction tree T, they are labelled by axiomatic sequences, and the corresponding sequents are axiomatic too. We proceed upwards in T by replacing each downward application of an R-S rule by an upwards application of the corresponding GS' rule.

The induction is finished at the root of T, which is labelled by the sequence $\neg\Gamma, \Delta$. Thus the sequent: $(\neg\Gamma, \Delta)^- \to (\neg\Gamma, \Delta)^+$ can be derived in GS', but this sequent need not be the original sequent $\Gamma \to \Delta$. The possible difference concerns the negative formulae: a formula $\neg\phi \in \neg\Gamma, \Delta$, may figure in the original sequent as ϕ on the left of '$\to$' or as $\neg\phi$ on the right. To obtain the original sequent from the above one, the required transformations can be performed using the added swapping rule (IV+) (and, possibly, (IV–)). $\Box$

The system GS" given below is a slightly simplified version of GS'. Each rule has the same number as the respective rule in the R-S system from which it was obtained.

Axioms

(I) $\Gamma \to x \prec x, \Delta \ : x \in X$ (II) $\Gamma, \gamma \to \gamma, \Delta$ (III) $\Gamma, \mathcal{E} \to t_s \prec t'_s, \Delta$

Replacement rules

	$+$	$-$
(IV)	$\dfrac{\Gamma, \gamma \to \Delta}{\Gamma \to \Delta, \neg\gamma}$	$\dfrac{\Gamma \to \Delta, \gamma}{\Gamma, \neg\gamma \to \Delta}$
(V)	$\dfrac{\Gamma \to \Delta, \gamma, \phi}{\Gamma \to \Delta, \gamma \vee \phi}$	$\dfrac{\Gamma, \gamma \to \Delta \mid \Gamma, \phi \to \Delta}{\Gamma, \gamma \vee \phi \to \Delta}$
(VI)	$\dfrac{\Gamma \to \Delta, \gamma \mid \Gamma \to \Delta, \phi}{\Gamma \to \Delta, \gamma \wedge \phi}$	$\dfrac{\Gamma, \gamma, \phi \to \Delta}{\Gamma, \gamma \wedge \phi \to \Delta}$
(VII)	$\dfrac{\Gamma, x \prec t \to \Delta, x \prec t'}{\Gamma \to \Delta, t \prec t'}$ $t \notin X$, and $x \in X$ is fresh	$\dfrac{\Gamma \to \Delta, x \prec t \mid \Gamma, x \prec t' \to \Delta}{\Gamma, t \prec t' \to \Delta}$ $t \notin X$ and $x \in X$ arbitrary
(VIII)	$\dfrac{\Gamma \to \Delta, y \prec t \mid \Gamma \to \Delta, x \prec f(\ldots, y, \ldots)}{\Gamma \to \Delta, x \prec f(\ldots, t, \ldots)}$ where $y \in X$ arbitrary and $t \notin X$	$\dfrac{\Gamma, y \prec t, x \prec f(\ldots, y, \ldots) \to \Delta}{\Gamma, x \prec f(\ldots, t, \ldots) \to \Delta}$ where $y \in X$ is fresh and $t \notin X$
(IX)	$\dfrac{\Gamma \to \Delta, t \doteq x \mid \Gamma \to \Delta, t' \doteq x}{\Gamma \to \Delta, t \doteq t'}$ $t, t' \notin X$ and $x \in X$ arbitrary	$\dfrac{\Gamma, t_s \doteq x_s, t'_s \doteq x_s \to \Delta}{\Gamma, t_s \doteq t'_s \to \Delta}$ $t_s, t'_s \notin X$ and $x_s \in X$ is fresh
(X)	$\dfrac{\Gamma \to \Delta, t_s \prec x_s \mid \Gamma \to \Delta, x_s \prec t_s \mid \Gamma \to \Delta, \neg\mathcal{E}_s}{\Gamma \to \Delta, t_s \doteq x_s}$ where $x_s \in X$ and $t_s \neq x_s$	$\dfrac{\Gamma, t_s \prec x_s, x_s \prec t_s, \neg\mathcal{E}_s \to \Delta}{\Gamma, t_s \doteq x_s \to \Delta}$ where $x_s \in X$ and $t_s \neq x_s$
(XI)	$\dfrac{\Gamma \to \Delta, t_s \prec x_s \mid \Gamma \to \Delta, x_s \prec t_s \mid \Gamma \to \Delta, \neg\mathcal{E}_s}{\Gamma \to \Delta, x_s \doteq t_s}$ where $x_s \in X$ and $t_s \neq x_s$	$\dfrac{\Gamma, t_s \prec x_s, x_s \prec t_s, \neg\mathcal{E}_s \to \Delta}{\Gamma, x_s \doteq t_s \to \Delta}$ where $x_s \in X$ and $t_s \neq x_s$
(AX)		$\dfrac{\Gamma, \gamma[\overline{y}_s/\overline{x}_s] \to \Delta}{\Gamma, ![\gamma] \to \Delta}$ where $\overline{y}_s$ arbitrary

Expansion rules

(XIV) $$\dfrac{\Gamma, y \prec f(\overline{z}) \to \Delta}{\Gamma, y \prec x, x \prec f(\overline{z}) \to \Delta}$$
(sound for arbitrary $x \in X$)

(XV) $$\dfrac{\Gamma, x \prec f(\ldots, y, \ldots) \to \Delta}{\Gamma, y \prec z, x \prec f(\ldots, z, \ldots) \to \Delta}$$
(sound for arbitrary $z \in X$)

(XVI) $$\dfrac{\Gamma, \mathcal{E}_s, x_{s'} \prec f(\ldots, y_s, \ldots), \mathcal{E}_{s'} \to \Delta}{\Gamma, \mathcal{E}_s, x_{s'} \prec f(\ldots, y_s, \ldots) \to \Delta}$$

(XII) $$\dfrac{\Gamma, x \prec y \to \Delta}{\Gamma, y \prec x \to \Delta}$$

(XIII) $$\dfrac{\Gamma, y \prec z \to \Delta}{\Gamma, y \prec x, x \prec z \to \Delta}$$

Theorem 4.5 *The system GS" given above is equivalent to the system GS', in particular, GS" is sound and complete, i.e., for any sequent* $\psi : \vdash_{GS''} \psi \iff \vdash_{GS'} \psi$.

PROOF. The replacement rules in GS" are essentially the same as in GS', i.e., are obtained directly by the translation of the respective R-S* rules. The only difference is that the GS" rules (VII–), (VIII+), (IX+), and the axiom rule (AX), i.e., the rules involving a choice of an arbitrary variable, do not repeat the active formulae (from the conclusion in the premisse(s)). This does not change the power of the system since the repetition of the active formulae (in R-S* and GS') can be now simulated by an immediate choice ("guessing") of the appropriate variable.

The new replacement rule, (IV–), was commented in the proof of completeness theorem 4.4, and is present in both systems GS' and GS".

The remaining expansion rules are simplified slightly in GS" by dropping some of the formulae from the premisses. (Thus, they are not invertible, though obviously sound.) For instance, following the translation schema from lemma 4.3, the symmetry rule in GS' looks as (XII') $\dfrac{\Gamma, x \prec y, y \prec x \to \Delta}{\Gamma, y \prec x \to \Delta}$, while in GS" it became (XII) $\dfrac{\Gamma, x \prec y \to \Delta}{\Gamma, y \prec x \to \Delta}$. However, each is admissible given the other, given admissibility of weakening rules (W) which follows by standard argument.

$$
\begin{array}{c|c}
\text{(W)+(XII')} \Rightarrow \text{(XII)} & \text{(XII)} \Rightarrow \text{(XII')} \\
\hline
\begin{array}{r}
\Gamma, x \prec y \to \Delta \\
\text{(W)}\quad \overline{\Gamma, y \prec x, x \prec y \to \Delta} \\
\text{(XII')}\quad \overline{\Gamma, y \prec x \to \Delta}
\end{array}
&
\begin{array}{l}
\Gamma, y \prec x, x \prec y \to \Delta \\
\overline{\Gamma, y \prec x, y \prec x \to \Delta} \quad \text{(XII)} \\
= \Gamma, y \prec x \to \Delta
\end{array}
\end{array}
$$

Equivalence of the other expansion rules from GS' and GS" is shown by similarly simple and entirely analogous derivations.

Weakening rules (W) are admissible in GS' (and in R-S*) by the standard argument. Consider the rule (W) $\dfrac{\Gamma \to \Delta}{\Gamma, \Lambda \to \Delta}$ and assume that $\Gamma \to \Delta$ is derivable in GS'. Each leaf of its proof tree T, which is an axiomatic sequent $\Gamma' \to \Delta'$, can be extended with Λ to $\Gamma', \Lambda \to \Delta'$, yielding again an axiomatic sequent. Propagating Λ's across the whole tree T will give a proof for $\Gamma, \Lambda \to \Delta$. (Possibly, the names of fresh variables at some nodes may need to be chosen differently in order not to clash with the names of variables in Λ.) By completeness of GS', the rule is admissible. □

Summarizing our results (the above theorem 4.5, soundness and completeness of GS' from theorem 4.4, we obtain a counterpart of the corollary 3.13 for the system GS":

Corollary 4.6 *For any specification $\Psi = \{\psi_1, \ldots, \psi_n\}$ and sequent ψ we have that:*

$$\Psi \models_* \psi \iff \vdash_{GS''} ![\mathsf{tr}(\psi_1)], \ldots, ![\mathsf{tr}(\psi_n)] \to \mathsf{tr}(\psi)$$

4.1 The final Gentzen system GS

Assuming that all our sequents are as indicated in the specifications, i.e., contain only atomic formulae, and observing that function tr (def. 3.4) introduces only disjunctions, the above corollary 4.6 holds also when we remove from GS" both rules (VI). We now perform a final transformation to obtain a "pure" sequent calculus for specifications, i.e., one operating only on sequents of atomic formuale and allowing to derive such sequents from specifications without any translation nor axiom rules.

For the sake of example, let our specification contain only one sequent, $\Psi = \{\gamma \to \delta\}$. To derive from it $\Gamma \to \Delta$, we would try to prove $![\neg\gamma \vee \delta], \Gamma \to \Delta$ which, applying the rule (AX), amounts to:

$$\frac{(\neg\gamma \vee \delta)[\overline{y}/\overline{x}], \Gamma \to \Delta}{![\neg\gamma \vee \delta], \Gamma \to \Delta}\ \overline{y} \text{ arbitrary} \tag{2}$$

where $\overline{y}$'s match the respective variables $\overline{x}$ from $\gamma \to \delta$. Applying the rules for disjunction (V) and for negation (IV–) in the antecedent of a sequent, we will end up with the assumptions as indicated in the following:

$$\frac{\delta[\overline{y}/\overline{x}], \Gamma \to \Delta \mid \Gamma \to \Delta, \gamma[\overline{y}/\overline{x}]}{![\neg\gamma \vee \delta], \Gamma \to \Delta}\ \overline{y} \text{ arbitrary} \tag{3}$$

All the assumptions are now sequents and this illustrates the idea of the final step. We are interested in proving statements of the form $\Psi \vdash \psi$, where all involved sequents are of the simple form $\gamma_1,\ldots,\gamma_n \to \delta_1,\ldots,\delta_m$, with all γ_i, δ_j being atomic inclusions or equalities. We can thus remove the rules for treating connectives, (IV), (V) and (VI), as well as the axiom rule (AX). We precede all assumptions and conclusions of the rules by $\Psi \vdash \ldots$ and add the rules of *specific cut*, [10], for each non-logical axiom $\gamma_1,\ldots,\gamma_n \to \delta_1,\ldots,\delta_m \in \Psi$:

$$\text{(SPC)}\quad \frac{\Psi \vdash \Gamma \to \Delta, \gamma_1' \mid \ldots \mid \Psi \vdash \Gamma \to \Delta, \gamma_n' \mid \Psi \vdash \Gamma, \delta_1' \to \Delta \mid \ldots \mid \Psi \vdash \Gamma, \delta_m' \to \Delta}{\Psi \vdash \Gamma \to \Delta}$$

where the primed versions denote uniform, arbitrary renaming of variables occurring in the involved axiom $\gamma_1,\ldots,\gamma_n \to \delta_1,\ldots,\delta_m \in \Psi$.

The possible simplification for the proofs by hand comes from the fact that we now do not have to write and carry around (the translations of) all the axioms of the specification, but can apply the rule (SPC) only for the needed axioms. In addition, of course, we no longer need to consider any other kinds of formulae or sequents and their translations, but only those consisting only of atomic formulae, as prescribed by the format of specifications.

Observe that, as argued in [10], the specific cut rules are significantly more manageable than the general cut. In fact, the "undecidability" of such rules (applied bottom-up) is essentially of the same kind as that of the axiom rules (AX) and concerns only the choice of the appropriate variable names.

The rules of the resulting system GS are given below. (Since we now consider only atomic formulae in the sequents, we have moved $\neg\mathcal{E}$ along the "$\to$" and replaced it with $\mathcal{E}$.) We can not claim the equivalence of GS" and GS, since the latter does not allow any formulae with axioms. However, taking into account the restrictions on such formulae we have put in GS" (only $\neg ![\ldots]$ occurring only at the outermost level, with the exception of one formula, corresponding to the sequent we are proving), the above remarks make it obvious that

$$\vdash_{GS''} ![\mathsf{tr}(\psi_1)],\ldots,![\mathsf{tr}(\psi_n)] \to \mathsf{tr}(\psi) \iff \{\psi_1,\ldots,\psi_n\} \vdash_{GS} \psi,$$

for any sequents $\psi_1,\ldots,\psi_n,\psi$ over atomic formulae. Indeed, if there is a proof in GS" involving an application of (AX), as in (2), then, moving "bottom-up", it must split the tree into branches for separate disjuncts (of each $\mathsf{tr}(\psi_i)[\overline{y}/\overline{x}]$) before processing the involved disjuncts themselves. Hence it must pass through nodes as given in the assumptions of (3). Except for the superficial differences of syntax, the rule (SPC) mimics exactly transition to such nodes. On the other hand, the rule is obviously sound (with the interpretation of $\{\psi_1\ldots\psi_n\} \vdash \psi$ as $\models ![\mathsf{tr}(\psi_1)],\ldots,![\mathsf{tr}(\psi_n)] \to \mathsf{tr}(\psi)$), and hence it is admissible in GS".

We thus obtain the calculus GS for deriving consequences of specifications, which does not require any transformation of the involved sequents, and the following theorem follows.

Theorem 4.7 *The system GS given below is sound and complete, i.e., for any specification Ψ and sequent ψ (involving only atomic formulae): $\Psi \models_* \psi \iff \Psi \vdash_{GS} \psi$.*

Axioms

(I) $\Psi \vdash \ \Gamma \to x \prec x, \Delta \ : x \in X$ (II) $\Psi \vdash \Gamma, \gamma \to \gamma, \Delta$ (III) $\Psi \vdash \ \Gamma, \mathcal{E} \to t_s \prec t'_s, \Delta$

Replacement rules

	+	−
(VII)	$\dfrac{\Psi \vdash \Gamma, x \prec t \to \Delta, x \prec t'}{\Psi \vdash \Gamma \to \Delta, t \prec t'}$ $t \notin X$, and $x \in X$ is fresh	$\dfrac{\Psi \vdash \Gamma \to \Delta, x \prec t \mid \Psi \vdash \Gamma, x \prec t' \to \Delta}{\Psi \vdash \Gamma, t \prec t' \to \Delta}$ $t \notin X$ and $x \in X$ arbitrary
(VIII)	$\dfrac{\Psi \vdash \Gamma \to \Delta, y \prec t \mid \Psi \vdash \Gamma \to \Delta, x \prec f(\ldots, y, \ldots)}{\Psi \vdash \Gamma \to \Delta, x \prec f(\ldots, t, \ldots)}$ where $y \in X$ arbitrary and $t \notin X$	$\dfrac{\Psi \vdash \Gamma, y \prec t, x \prec f(\ldots, y, \ldots) \to \Delta}{\Psi \vdash \Gamma, x \prec f(\ldots, t, \ldots) \to \Delta}$ where $y \in X$ is fresh and $t \notin X$
(IX)	$\dfrac{\Psi \vdash \Gamma \to \Delta, t \doteq x \mid \Psi \vdash \Gamma \to \Delta, t' \doteq x}{\Psi \vdash \Gamma \to \Delta, t \doteq t'}$ $t, t' \notin X$ and $x \in X$ arbitrary	$\dfrac{\Psi \vdash \Gamma, t_s \doteq x_s, t'_s \doteq x_s \to \Delta}{\Psi \vdash \Gamma, t_s \doteq t'_s \to \Delta}$ $t_s, t'_s \notin X$ and $x_s \in X$ is fresh
(X)	$\dfrac{\Psi \vdash \Gamma \to \Delta, t_s \prec x_s \mid \Psi \vdash \Gamma \to \Delta, x_s \prec t_s \mid \Psi \vdash \Gamma, \mathcal{E}_s \to \Delta}{\Psi \vdash \Gamma \to \Delta, t_s \doteq x_s}$ where $x_s \in X$ and $t_s \neq x_s$	$\dfrac{\Psi \vdash \Gamma, t_s \prec x_s, x_s \prec t_s \to \Delta, \mathcal{E}_s}{\Psi \vdash \Gamma, t_s \doteq x_s \to \Delta}$ where $x_s \in X$ and $t_s \neq x_s$
(XI)	$\dfrac{\Psi \vdash \Gamma \to \Delta, t_s \prec x_s \mid \Psi \vdash \Gamma \to \Delta, x_s \prec t_s \mid \Psi \vdash \Gamma, \mathcal{E}_s \to \Delta}{\Psi \vdash \Gamma \to \Delta, x_s \doteq t_s}$ where $x_s \in X$ and $t_s \neq x_s$	$\dfrac{\Psi \vdash \Gamma, t_s \prec x_s, x_s \prec t_s \to \Delta, \mathcal{E}_s}{\Psi \vdash \Gamma, x_s \doteq t_s \to \Delta}$ where $x_s \in X$ and $t_s \neq x_s$

Specific cut rules

(SPC) $$\frac{\Psi \vdash \Gamma \to \Delta, \gamma'_1 \ldots \Psi \vdash \Gamma \to \Delta, \gamma'_n \mid \Psi \vdash \Gamma, \delta'_1 \to \Delta \ldots \Psi \vdash \Gamma, \delta'_m \to \Delta}{\Psi \vdash \Gamma \to \Delta}$$

for each axiom $\gamma_1, \ldots, \gamma_n \to \delta_1, \ldots, \delta_m \in \Psi$, with arbitrary renaming $'$ of variables

Expansion rules

(XIV) $$\frac{\Psi \vdash \Gamma, y \prec f(\overline{z}) \to \Delta}{\Psi \vdash \Gamma, y \prec x, x \prec f(\overline{z}) \to \Delta}$$

(sound for arbitrary $x \in X$)

(XII) $$\frac{\Psi \vdash \Gamma, x \prec y \to \Delta}{\Psi \vdash \Gamma, y \prec x \to \Delta}$$

(XV) $$\frac{\Psi \vdash \Gamma, x \prec f(\ldots, y, \ldots) \to \Delta}{\Psi \vdash \Gamma, y \prec z, x \prec f(\ldots, z, \ldots) \to \Delta}$$

(sound for arbitrary $z \in X$)

(XIII) $$\frac{\Psi \vdash \Gamma, y \prec z \to \Delta}{\Psi \vdash \Gamma, y \prec x, x \prec z \to \Delta}$$

(XVI) $$\frac{\Psi \vdash \Gamma, \mathcal{E}_s, x_{s'} \prec f(\ldots, y_s, \ldots), \mathcal{E}_{s'} \to \Delta}{\Psi \vdash \Gamma, \mathcal{E}_s, x_{s'} \prec f(\ldots, y_s, \ldots) \to \Delta}$$

We finish with an example showing the simplification in proofs in GS as compared to R-S*.

Example 4.8 *Suppose that the specification Ψ has two axioms*

1. $f(x) \prec c \rightarrow g(x) \doteq d$ *and*
2. $g(x) = d \rightarrow h(x) \prec a$.

We want to prove $\Psi \vdash f(x) \prec c \rightarrow h(x) \prec a$. We first give the proof in the Gentzen calculus: If a branch terminates, the axiomatic subsequences are underlined. We drop sort subscripts.

$$\frac{\Psi \vdash f(x) \prec c, \underline{g(x) \doteq d} \rightarrow h(x) \prec a, \underline{g(x) \doteq d} \mid \Psi \vdash f(x) \prec c, g(x) \doteq d, \underline{h(x) \prec a} \rightarrow \underline{h(x) \prec a}}{i = \quad \Psi \vdash f(x) \prec c, g(x) \doteq d \rightarrow h(x) \prec a} \; \textit{(SPC) ax.2}$$

$$\frac{\Psi \vdash \underline{f(x) \prec c} \rightarrow h(x) \prec a, \underline{f(x) \prec c} \quad \mid \; i}{\Psi \vdash f(x) \prec c \rightarrow h(x) \prec a} \; \textit{(SPC) ax.1}$$

And the proof in the R-S calculus – the active formulae are in boldface.*

$$\frac{![(\neg(\mathbf{f(x)} \prec \mathbf{c}) \vee \mathbf{g(x)} \doteq \mathbf{d})], ![(\neg(g(x) \doteq d) \vee h(x) \prec a)], \neg(f(x) \prec c) \vee h(x) \prec a}{\neg(\neg(f(x) \prec c) \vee g(x) \doteq d), ![(\neg(g(x) \doteq d) \vee h(x) \prec a)], \neg(f(x) \prec c) \vee h(x) \prec a} \; \textit{(AX)}$$

$$\frac{\neg(\neg(f(x) \prec c) \vee g(x) \doteq d), ![(\neg(\mathbf{g(x)} \doteq \mathbf{d}) \vee \mathbf{h(x)} \prec \mathbf{a})], \neg(f(x) \prec c) \vee h(x) \prec a}{\neg(\neg(f(x) \prec c) \vee g(x) \doteq d), \neg(\neg(g(x) \doteq d) \vee h(x) \prec a), \neg(f(x) \prec c) \vee h(x) \prec a} \; \textit{(AX)}$$

$$\frac{\neg(\neg(f(x) \prec c) \vee g(x) \doteq d), \neg(\neg(g(x) \doteq d) \vee h(x) \prec a), \neg(\mathbf{f(x)} \prec \mathbf{c}) \vee \mathbf{h(x)} \prec \mathbf{a}}{\neg(\neg(f(x) \prec c) \vee g(x) \doteq d), \neg(\neg(g(x) \doteq d) \vee h(x) \prec a), \neg(f(x) \prec c), h(x) \prec a} \; \textit{(V+)}$$

$$\frac{\neg(\neg(\mathbf{f(x)} \prec \mathbf{c}) \vee \mathbf{g(x)} \doteq \mathbf{d}), \neg(\neg(g(x) \doteq d) \vee h(x) \prec a), \neg(f(x) \prec c), h(x) \prec a}{\dfrac{\neg\neg(\mathbf{f(x)} \prec \mathbf{c}), \neg(\neg(g(x) \doteq d) \vee h(x) \prec a), \neg(f(x) \prec c), h(x) \prec a}{\underline{f(x) \prec c}, \neg(\neg(g(x) \doteq d) \vee h(x) \prec a), \underline{\neg(f(x) \prec c)}, h(x) \prec a} \; \textit{(IV–)} \quad \mid \; i} \; \textit{(V–)}$$

$$\frac{i = \quad \neg g(x) \doteq d, \neg(\neg(\mathbf{g(x)} \doteq \mathbf{d}) \vee \mathbf{h(x)} \prec \mathbf{a}), \neg(f(x) \prec c), h(x) \prec a}{j \quad \mid \quad \neg g(x) \doteq d, \underline{\neg h(x) \prec a}, \neg(f(x) \prec c), \underline{h(x) \prec a}} \; \textit{(V–)}$$

$$\frac{j = \quad \neg g(x) \doteq d, \neg\neg(\mathbf{g(x)} \doteq \mathbf{d}), \neg(f(x) \prec c), h(x) \prec a}{\underline{\neg g(x) \doteq d}, \underline{g(x) \doteq d}, \neg(f(x) \prec c), h(x) \prec a} \; \textit{(IV–)}$$

5 Conclusions

We have applied the technique of Rasiowa-Sikorski [11] for designing sound, complete and cut-free logics for reasoning about multialgebras. We hope that the detailed proofs, included in this paper, may draw more attention to this elegant and powerful technique, and faciliate its broader applications. More details on and applications of this technique can be found in [1, 5, 4].

As compared to the most closely related work which also used this technique, [2], the main difference is the presence of the new predicate, $\doteq$, which was not included in the language of [2]. We have argued why this predicate is relevant and useful, especially, for writing specifications of nondeterministic data types, and we have shown how (non-)empty carriers can be treated using this predicate instead of quantifiers needed in [2]. Furthermore, the logic from [2] allows one to derive only tautologies but not logical consequences of sets of given, non-logical axioms. We have elaborated the possibility (only implicit in [2]) of extending logic for such purpose, by providing the required translation schema. Then, we have shown how this translation schema (as well as rules for connectives and axioms), needed to handle non-logical axioms in the R-S* system (and in [2]), can be removed and replaced by the specific cut rules, inspired by [10]. The resulting system can be used directly, without any intermediary transformations, for deriving consequences from specifications, that is, sequents from sets of sequents, and the obtained simplifications were illustrated by an example.

The unique decomposition tree which provides a proof strategy and has been identified for the introduced logics R-S and R-S*, following [11], is a natural candidate for a possible implementation and we expect that such an implementation will become available in not too far future.

References

[1] Arnon Avron and Beata Konikowska. Decomposition systems for gödel-dummett logics. *Studia Logica, special issue: Analytic Proof Techniques*, 2000.

[2] Marcin Białasik and Beata Konikowska. Reasoning with first-order nondeterministic specifications. *Acta Informatica*, 36:357–403, 1999.

[3] Heinrich Hussmann. *Nondeterminism in Algebraic Specifications and Algebraic Programs.* Birkhäuser, 1993.

[4] Beata Konikowska. Rasiowa-sikorski deduction systems: a handy tool for computer science logic. In J. Fiadeiro, editor, *Recent Trends in Algebraic Specification Techniques*, volume 1589 of *LNCS*. Springer, 1999.

[5] Beata Konikowska. Rasiowa-sikorski deduction systems in computer science applications. Technical Report 916, IPI PAN, Warsaw, 2000. [to appear in Theoretical Computer Science, special issue: Algebraic Development Techniques].

[6] Yngve Lamo and Michał Walicki. Modeling partiality by nondeterminism - from abstract specifications to flexible error handling. Technical Report 178, Depeartment of Informatics, University of Bergen, 1999.

[7] Yngve Lamo and Michał Walicki. The institution of multialgebras. Technical Report 209, Depeartment of Informatics, University of Bergen, 2000.

[8] Yngve Lamo and Michał Walicki. Modeling partiality by nondeterminism. In N. Callaos, J. M. Pineda, and M. Sanchez, editors, *Proceedings of SCI/ISAS 2001*, volume I. Orlando, FL, 2001.

[9] Yngve Lamo and Michał Walicki. Logic with equality for multialgebras. Technical Report 227, Depeartment of Informatics, University of Bergen, 2002.

[10] Aida Pliuškievičienė. Specialization of the use of axioms for deduction search in axiomatic theories with equality. *J. Soviet Math.*, 1, 1973.

[11] H. Rasiowa and R. Sikorski. *The Mathematics of Metamatemathics*. PWN [Polish Scientific Publishers], 1963.

[12] Michał Walicki and Sigurd Meldal. A complete calculus for the multialgebraic and functional semantics of nondeterminism. *ACM TOPLAS*, 17(2), March 1995.

[13] Michał Walicki and Sigurd Meldal. Multialgebras, power algebras and complete calculi of identities and inclusions. In *Recent Trends in Algebraic Specification Techniques*, volume 906 of *LNCS*. Springer, 1995.

[14] Michał Walicki and Sigurd Meldal. Algebraic approaches to nondeterminism-an overview. *ACM Computing Surveys*, 29, 1997.