Theoretical Computer Science 30 (1984) 1–48
North-Holland

PROVING PROGRAM INCLUSION USING HOARE'S LOGIC

J.A. BERGSTRA*
Department of Computer Science, University of Leiden, 2300 RA Leiden, The Netherlands

J.W. KLOP
Department of Computer Science, Mathematical Centre, 1098 SJ Amsterdam, The Netherlands

Communicated by C. Böhm
Received December 1981
Revised February 1983

Abstract. We explore conservative refinements of specifications. These form a quite appropriate framework for a proof theory for program inclusion based on a proof theory for program correctness.
We propose two formalized proof methods for program inclusion and prove these to be sound. Both methods are incomplete but seem to cover most natural cases.

Key words. Data type specification, program correctness, conservative refinement, program inclusion, program equivalence, prototype proof, logical completion.

Contents

Introduction

This paper aims at a detailed study of program equivalence, seen from the point of view of Hoare's logic for program correctness. Because program inclusion is just halfway program equivalence we can safely restrict our attention to program

* Present affiliation: Department of Computer Science, Mathematical Centre, 1098 SJ Amsterdam, The Netherlands.

0304-3975/84/$3.00 © 1984, Elsevier Science Publishers B.V. (North-Holland)
Bibliotheek

inclusion. Moreover, this has the advantage of connecting closely to the theory of programming using stepwise refinements as described in [2].

Our work can be seen as belonging to the subject of axiomatic semantics for programs. Its novelty lies in the precise mathematical analysis of the situation, in addition to a rather strict adherence to first order proof systems and first order semantics for data type specifications.

Deriving program equivalence from program correctness properties is, of course, not a new idea. It occurs in compiler correctness proofs (for instance, [16, 23]) as well as in the general theory of program correctness [15].

Because of our interest in a proper theoretical analysis, we try to minimize the semantical problems by working with **while**-programs only; this by no means trivializes the problem.

In the sequel of this Introduction an intuitive account is given of the key definitions that underly the paper.

Intuition

Suppose that for $S_1, S_2 \in \mathcal{WP}(\Sigma)$ we have

(i) $\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2$ (semantical inclusion)

and that we wish to prove this fact. Now obviously, (i) implies

(ii) $\mathrm{Alg}(\Sigma, E) \models \{p\}\, S_2 \{q\} \Rightarrow \mathrm{Alg}(\Sigma, E) \models \{p\}\, S_1 \{q\}$ for all $p, q \in L(\Sigma)$.

However, there is no reason to expect that the reverse implication (ii) $\Rightarrow$ (i) will hold, since (ii) states only roughly that $S_1 \sqsubseteq S_2$, where 'roughly' refers to the limited expressive power of $L(\Sigma)$. (In fact, Remark 7.8(2) shows that indeed (ii) $\not\Rightarrow$ (i).) Now consider

(iii) $\forall (\Sigma', E') \geq (\Sigma, E)\ \forall p, q \in L(\Sigma')$

$$\mathrm{Alg}(\Sigma', E') \models \{p\}\, S_2 \{q\} \Rightarrow \mathrm{Alg}(\Sigma', E') \models \{p\}\, S_1 \{q\}.$$

Clearly (i) $\Rightarrow$ (iii) $\Rightarrow$ (ii). (For (i) $\Rightarrow$ (iii), note that if $(\Sigma', E') \geq (\Sigma, E)$, then the reducts of (Σ', E')-algebras to Σ form a subset of $\mathrm{Alg}(\Sigma, E)$; hence $\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2 \Rightarrow \mathrm{Alg}(\Sigma', E') \models S_1 \sqsubseteq S_2$.)

In fact, we will restrict our attention to a subclass of all refinements ($\geq$) of (Σ, E), namely to the *conservative* refinements ($\trianglerighteq$) of (Σ, E), for reasons which will be clear later. So consider

(iv) $\forall (\Sigma', E') \trianglerighteq (\Sigma, E)\ \forall p, q \in L(\Sigma')$

$$\mathrm{Alg}(\Sigma', E') \models \{p\}\, S_2 \{q\} \Rightarrow \mathrm{Alg}(\Sigma', E') \models \{p\}\, S_1 \{q\}.$$

Now we have (i) $\Rightarrow$ (iii) $\Rightarrow$ (iv) $\Rightarrow$ (ii); and it turns out that (iv) $\Rightarrow$ (i) (see Remark 7.8(3)). The conclusion is that one can treat the 'semantical' inclusion (i) by considering only first order properties of S_1, S_2 (i.e., asserted programs

$\{p\}\, S_i\, \{q\}$, $i = 1, 2$), provided one is willing to consider not only (Σ, E), but all its (conservative) refinements.

This observation prepares the way for an approach via Hoare's logic of proving asserted programs. First of all, define

$$\text{(v)} \quad S_i \sqsubseteq_{\mathrm{HL}(\Sigma,E)} S_2 \ \text{ iff } \ \forall p, q (L(\Sigma))\, (\mathrm{HL}(\Sigma, E) \vdash \{p\}\, S_1\, \{q\}$$
$$\Rightarrow \mathrm{HL}(\Sigma, E) \vdash \{p\}\, S_1\, \{q\}) \ \ \text{(proof-theoretical inclusion)}$$

and consider

$$\text{(vi)} \quad \forall (\Sigma', E') \trianglerighteq (\Sigma, E) \ \ S_1 \sqsubseteq_{\mathrm{HL}(\Sigma', E')} S_2 \quad \text{(derivable inclusion)}$$

the proof-theoretical analogue of (iv). Indeed, it will turn out that this 'derivable inclusion', written as $\mathrm{HL}(\Sigma, E) \vdash S_1 \sqsubseteq S_2$, implies the semantical inclusion (i). This is our first 'proof system' for proving semantical inclusion; we will prove that (vi), as a relation of S_1, S_2, is semi-decidable in E.

One more remark about why it is natural to consider (vi), in casu the quantification over all conservative refinements. The first reason of considering all (conservative) refinements of (Σ, E) is that, only then, one is able to give as refined as possible first order descriptions of $S_1 \sqsubseteq S_2$. This holds already on the semantical level. Moreover, in (vi) there is another reason: to *prove* $\{p\}\, S\, \{q\}$ we need invariants for the **while**-loops in S. It may be the case that these invariants cannot yet be expressed in the present specification, so we have to go 'higher-up'. If one attributes a defining power to statements S, namely to define the invariants of the **while**-loops, then one could say that the defining power of $S \in \mathcal{WP}(\Sigma)$ is sometimes ahead of that of the assertion language $L(\Sigma)$.

Of course, the proof system given by (vi) is sound, i.e., (vi) $\Rightarrow$ (i); otherwise it did not deserve the name. Some simple program inclusions that are in its scope, are program equivalences like 'loop-unwinding', and the kind of program equivalences considered in [20]. However, this proof system is not yet complete. In order to prove the semantical inclusion (i) it is sufficient that (see Fig. 1)

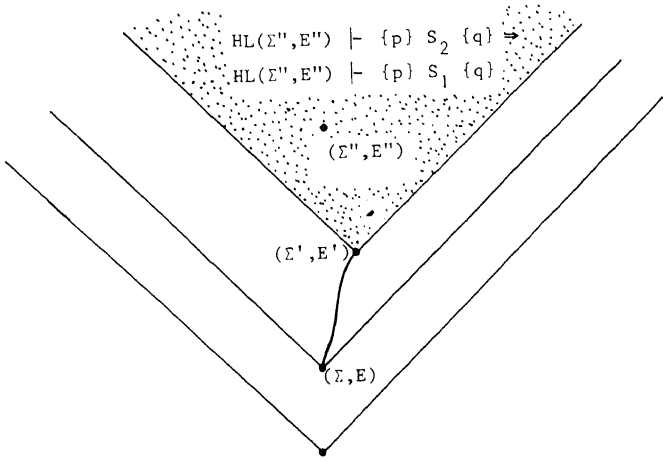


Fig. 1. Partial order of conservative refinements.

(vii) $\exists(\Sigma', E') \trianglerighteq (\Sigma, E)\ \forall(\Sigma'', E'') \trianglerighteq (\Sigma', E')\ S_1 \sqsubseteq_{\mathrm{HL}(\Sigma'', E'')} S_2$.

(*Notation*: $\mathrm{HL}(\Sigma, E) \Vdash S_1 \sqsubseteq S_2$; in words: forced inclusion.)

The reason that (vii) $\Rightarrow$ (i) is a simple consequence of the invariance of semantical inclusion (i), i.e., if $(\Sigma', E') \trianglerighteq (\Sigma, E)$ and $S_1, S_2 \in \mathcal{WP}(\Sigma)$, then

$$\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2 \Leftrightarrow \mathrm{Alg}(\Sigma', E') \models S_1 \sqsubseteq S_2.$$

(This does not hold for $\geq$ instead of $\trianglerighteq$.) So in order to prove $\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2$ it is sufficient to find some $(\Sigma', E') \trianglerighteq (\Sigma, E)$ where $\mathrm{Alg}(\Sigma', E') \models S_1 \sqsubseteq S_2$.

The proof system embodied by (vii) is stronger than that of the derivable inclusion (vi), and we will give some examples of program inclusion (which seem to have some practical interest, too) which require the extra strength of this last proof system.

Still, (vii) is not 'complete'—although it seems hard to find a non-pathological example of a program inclusion which is semantical (i), but which cannot be forced (vii). One can prove, however, that the following 'cofinal' inclusion is equivalent to semantical inclusion:

(viii) $\forall(\Sigma', E') \trianglerighteq (\Sigma, E)\ \exists(\Sigma'', E'') \trianglerighteq (\Sigma', E')\ S_1 \sqsubseteq_{\mathrm{HL}(\Sigma'', E'')} S_2$.

(The equivalence (i)$\Leftrightarrow$(viii) holds also when in (viii) $\trianglerighteq$ is replaced by $\geq$. However, for $\trianglerighteq$ we have (vii) $\Rightarrow$ (viii), not so for $\geq$.)

One could suspect that there is a multitude of such relations obtained by repeated alternating quantification $\forall\exists\forall\cdots$ from the basic relation $\sqsubseteq_{\mathrm{HL}(\Sigma,E)}$ (proof-theoretical inclusion). It is a pleasant surprise, suggesting the naturalness of the notions involved, that this possible hierarchy does in fact not exist, and that one has no more relations than in Fig. 2.

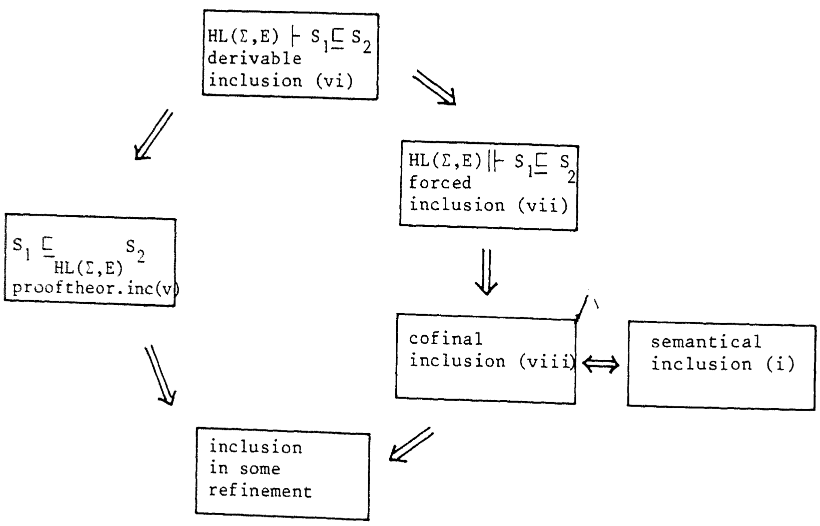


Fig. 2.

As we have seen, conservative refinements ($\trianglerighteq$) are more natural for this theory than general refinements ($\geq$). The technical reason is that for conservative refinements the 'Joint Refinement Property' holds, stating that (almost) every two refinements $(\Sigma_i, E_i) \trianglerighteq (\Sigma, E)$ can be refined to a common refinement $(\Sigma_3, E_3) \trianglerighteq (\Sigma_i, E_i)$ $(i = 1, 2)$. (This is in fact a strengthened version of the well-known Robinson Consistency Theorem.) Also for conservative refinements we have a useful upward and downward invariance of the properties

$$\mathrm{Alg}(\Sigma', E') \models \{p\}\, S\, \{q\} \quad \text{and} \quad \mathrm{Alg}(\Sigma', E') \models S_1 \sqsubseteq S_2 \quad \text{for } (\Sigma', E') \trianglerighteq (\Sigma, E).$$

This paper is built up as follows: in Section 1 some notions about logic, programs and Hoare's logic are given. Section 2 gives a criterion and a characterization of conservativity, and also Robinson's Consistency Theorem (our Corollary 2.6.2) is stated. Section 3 states Padoa's method (our Theorem 3.3) and gives some applications. Section 4 contains definitions of the various inclusions. In Section 5 we deal with the technical concept of protototype proofs, which will be basic for the proof systems in the sequel. In Section 6 a logical complete refinement is constructed for each specification. In Section 7 one of the main theorems is proved, establishing the existence of two proof systems for $\sqsubseteq$. In Section 8 we consider a prime example to yield more insight in the relations between the various inclusions. In Section 9 we will show that some additional information about the domains of S_1, S_2 can be converted to information about semantical and forced inclusion $S_1 \sqsubseteq S_2$.

1. Preliminaries

In this section we will collect the necessary basic definitions and facts from logic in general as well as Hoare's logic.

1.1. Preliminaries about programs and logic

The notions of first-order language, derivability ($\vdash$) and satisfiability ($\models$) are supposed to be well known and we repeat them merely to fix the notations and terminology used in the sequel.

In this paper we will exclusively deal with $\mathcal{WP}(\Sigma)$, the set of **while**-program S defined inductively as follows:

$$S ::= x := t \mid S_1 ; S_2 \mid \textbf{if } b \textbf{ then } S_1 \textbf{ else } S_2 \textbf{ fi} \mid \textbf{while } b \textbf{ do } S \textbf{ od},$$

where $t \in \mathrm{Ter}(\Sigma)$, the set of terms over the signature Σ, b is a boolean (i.e., quantifier-free) assertion $\in L(\Sigma)$, the first-order language determined by Σ. In general, assertions $\in L(\Sigma)$ will be denoted by p, q, r. The signature says what 'non-logical' symbols we are considering; here equality ($=$) is considered as a logical symbol. We also allow infinite signatures. For a further definition of signatures and specifications, see Definition 2.1. Note that the signature defined there is part of the alphabet of $L(\Sigma)$.

If (Σ, E) is a specification (see again Definition 2.1), the algebras (or models) in $\mathrm{Alg}(\Sigma, E)$ will be denoted by $\mathscr{A} = \langle A, \ldots \rangle$ where A is the underlying set of the algebraic structure $\mathscr{A}$.

We will need the following well-known fact.

1.1.1. Gödel's completeness theorem

$$(\Sigma, E) \vdash p \Leftrightarrow \mathrm{Alg}(\Sigma, E) \models p \quad \textit{for all } p \in L(\Sigma).$$

We will also need the following lemma.

1.1.2. Computation Lemma. *Let* $\boldsymbol{x} = x_1, \ldots, x_k$ *and* $\boldsymbol{y} = y_1, \ldots, y_k$. *Let* $S = S(\boldsymbol{x}) \in \mathscr{WP}(\Sigma)$ *(i.e., S contains precisely the variables* $\boldsymbol{x}$*).*

Then for all $n \in \mathbb{N}$ *there is a quantifier-free assertion* $\mathrm{Comp}_{S,n}(\boldsymbol{x}) = \boldsymbol{y}$ *in* $L(\Sigma)$ *such that, for every* $\mathscr{A} \in \mathrm{Alg}(\Sigma)$ *and all* $\boldsymbol{a}, \boldsymbol{b} \in A$,

$$\mathscr{A} \models \mathrm{Comp}_{S,n}(\underline{\boldsymbol{a}}) = \underline{\boldsymbol{b}} \Leftrightarrow |S(\boldsymbol{a})| \leq n \ \&\ S(\boldsymbol{a}) = \boldsymbol{b}.$$

Here $\underline{\boldsymbol{a}}, \underline{\boldsymbol{b}}$ are constant symbols denoting $\boldsymbol{a}, \boldsymbol{b}$ and $|S(\boldsymbol{a})|$ denotes the length of the computation of S on $\boldsymbol{a}$.

1.2. Preliminaries on Hoare's logic

Let $p, q \in L(\Sigma)$ and $S \in \mathscr{WP}(\Sigma)$. Then the syntactic object $\{p\}\ S\ \{q\}$ is called an *asserted program.* If $\mathscr{A} \in \mathrm{Alg}(\Sigma)$, we define

$$\mathscr{A} \models \{p\}\ S\ \{q\} \Leftrightarrow \forall \boldsymbol{a}, \boldsymbol{b} \in A\colon S\ (\boldsymbol{a}) \downarrow \ \&\ S(\boldsymbol{a}) = \boldsymbol{b} \Leftrightarrow (\mathscr{A} \models p(\underline{\boldsymbol{a}}) \to q(\underline{\boldsymbol{b}})).$$

Furthermore, we define

$$\mathrm{Alg}(\Sigma, E) \models \{p\}\ S\ \{q\} \Leftrightarrow \forall \mathscr{A} \in \mathrm{Alg}(\Sigma, E)\ \mathscr{A} \models \{p\}\ S\ \{q\}.$$

Hoare's logic w.r.t. (Σ, E) is a proof system designed to prove facts like $\mathrm{Alg}(\Sigma, E) \models \{p\}\, S\, \{q\}$. We will call this proof system $\mathrm{HL}(\Sigma, E)$. It has the following axioms and rules, by means of which we can derive asserted programs (notation: $\mathrm{HL}(\Sigma, E) \vdash \{p\}\, S\, \{q\}$):

(1) *Assignment axiom*:

$$\{p[t/x]\}\ x := t\ \{p\}$$

(2) *Composition rule*:

$$\frac{\{p\}\ S_1\ \{r\} \quad \{r\}\ S_2\ \{q\}}{\{p\}\ S_1\,;\, S_2\ \{q\}}$$

(3) *Conditional rule*:

$$\frac{\{p \wedge b\}\ S_1\ \{q\} \quad \{p \wedge \neg b\}\ S_2\ \{q\}}{\{p\}\ \textbf{if}\ b\ \textbf{then}\ S_1\ \textbf{else}\ S_2\ \textbf{fi}\ \{q\}}$$

(4) *Iteration rule*:

$$\frac{\{p \wedge b\}\, S\, \{p\}}{\{p\}\ \textbf{while}\ b\ \textbf{do}\ S\ \textbf{od}\ \{p \wedge \neg b\}}$$

(5) *Consequence rule*:

$$\frac{p \to p_1 \quad \{p_1\}\, S\, \{q_1\} \quad q_1 \to q}{\{p\}\, S\, \{q\}}$$

where $(\Sigma, E) \vdash p \to p_1$ and $(\Sigma, E) \vdash q_1 \to q$.

1.2.1. Lemma. $\mathrm{HL}(\Sigma, E)$ *is sound, i.e., for all* $p, S, q \in L(\Sigma)$:

$$\mathrm{HL}(\Sigma, E) \vdash \{p\}\, S\, \{q\} \Rightarrow \mathrm{Alg}(\Sigma, E) \models \{p\}\, S\, \{q\}.$$

Proof. For the proof, see, e.g., [13]. □

1.2.2. Definition. $\mathrm{HL}(\Sigma, E)$ is *logically complete*, if, for all $p, S, q \in L(\Sigma)$,

$$\mathrm{HL}(\Sigma, E) \vdash \{p\}\, S\, \{q\} \Leftrightarrow \mathrm{Alg}(\Sigma, E) \models \{p\}\, S\, \{q\}.$$

(In general, $\mathrm{HL}(\Sigma, E)$ is not logically complete. The notion of logical completeness is studied in [7].)

From the axioms and rules of $\mathrm{HL}(\Sigma, E)$ one can derive the following useful rules.

1.2.3. (i) *Conjunction rule*:

$$\frac{\{p_1\}\, S\, \{q_1\} \quad \{p_2\}\, S\, \{q_2\}}{\{p_1 \wedge p_2\}\, S\, \{q_1 \wedge q_2\}}$$

(ii) *Disjunction rule*: The same as (i) with $\wedge$ replaced by $\vee$.

(iii) *Invariance rule*: If the free variables in p are disjoint from the variables in S, then $\mathrm{HL}(\Sigma, E) \vdash \{p\}\, S\, \{p\}$

(iv) $\exists$*-rule*:

$$\frac{\{p\}\, S\, \{r\}}{\{\exists z\, p\}\, S\, \{r\}} \quad \text{provided } z \text{ does not occur in } S.$$

2. Conservative refinements

In this section we will collect some facts concerning the notion of *refinement* and, especially, *conservative* refinement. These notions will be of fundamental importance in the sequel. All the material in this section (and the next, on 'definability') is standard in Mathematical Logic and can be found (e.g.) in [24, 21]. For easier reference and to conform to our notations, we will give a fairly extensive survey of the subject. Since the arguments used in the proofs are relevant for the sequel, we have included some of the proofs.

2.1. Definition. (i) A *signature* Σ is a set of nonlogical symbols to be used in Predicate Logic. These may be constant, function or predicate symbols; the *arity* of a function or predicate symbol is the number of arguments it is supposed to have.

(E.g., $\Sigma = \{\underline{0}, S, P, <\}$ is a signature where $\underline{0}$ is a constant symbol, S and P are unary function symbols and $<$ is a binary predicate symbol.) $L(\Sigma)$ denotes the set of assertions in which only nonlogical symbols $\pi, \sigma \in \Sigma$ occur.

(ii) If $E \subseteq L(\Sigma)$, the pair (Σ, E) is called a *specification.*

(iii) $\mathrm{Alg}(\Sigma)$ is the class of all Σ-algebras.

(E.g., $\mathcal{A} = (\mathbb{N}, 0, s, p, k) \in \mathrm{Alg}(\Sigma)$, where Σ is as in the example above. Here 0 is a constant of $\mathcal{A}$, s and p are unary functions and k is a binary relation. We will also write $S^{\mathcal{A}}$ for the *interpretation* or *semantics* of S in $\mathcal{A}$, in casu s; for convenience we will often neglect to distinguish notationally the symbol from its interpretation.)

(iv) $\mathrm{Alg}(\Sigma, E)$ is the class of Σ-algebras $\mathcal{A}$ such that $\mathcal{A} \models E$.

(v) $\mathrm{Alg}(\Sigma, E) \models p$ means: for all $\mathcal{A} \in \mathrm{Alg}(\Sigma, E)$, $\mathcal{A} \models p$.

2.2. Definition. (i) If $\Sigma' \supseteq \Sigma$ and $\bar{E}' \supseteq \bar{E}$ we write $(\Sigma', E') \geq (\Sigma, E)$ and call (Σ', E') a *refinement* of (Σ, E). Here $\bar{E} = \{p \in L(\Sigma) \mid E \vdash p\}$. We will always suppose that E, E' are consistent.

(ii) If (Σ', E') is finite (i.e., both Σ' and E' are finite), then we write $(\Sigma \cup \Sigma', E \cup E') \geq_{\mathrm{f}} (\Sigma, E)$.

(iii) Let $\mathcal{A}$ be some algebra. Then $\Sigma_{\mathcal{A}}$ is the *signature of* $\mathcal{A}$ and $E_{\mathcal{A}}$ is the *theory of* $\mathcal{A}$: $E_{\mathcal{A}} = \{p \in L(\Sigma_{\mathcal{A}}) \mid \mathcal{A} \models p\}$. Note that $\mathcal{A} \models p \Leftrightarrow \mathrm{Alg}(\Sigma_{\mathcal{A}}, E_{\mathcal{A}}) \models p$.

(iv) Let (Σ, E) be a specification. Then E is *complete* if $\forall p \in L(\Sigma), E \vdash p$ or $E \vdash \neg p$.

2.3. Definition. (i) Let $(\Sigma', E') \geq (\Sigma, E)$ be a refinement such that: $\forall p \in L(\Sigma) E' \vdash p \Leftrightarrow E \vdash p$. In other words, such that $\overline{E'} \cap L(\Sigma) = \bar{E}$. Then this refinement is called *conservative* over (Σ, E). (So a conservative refinement does not yield more theorems in the 'original' language $L(\Sigma)$.)

Notation: $(\Sigma', E') \trianglerighteq (\Sigma, E)$

(ii) $(\Sigma', E') \trianglerighteq_{\mathrm{f}} (\Sigma, E) \Leftrightarrow (\Sigma', E') \trianglerighteq (\Sigma, E)$ & $(\Sigma', E') \geq_{\mathrm{f}} (\Sigma, E)$.

2.3.1. Remark. Note that if E is complete, $(\Sigma', E') \geq (\Sigma, E) \Rightarrow (\Sigma', E') \trianglerighteq (\Sigma, E)$.

2.4. Definition. Let $\Sigma' \supseteq \Sigma$.

(i) If (Σ', E') is a specification, then the *restriction* of (Σ', E') to the signature Σ is (Σ, E) where $E = \overline{E'} \cap L(\Sigma)$.

We write $\rho_{\Sigma}^{\Sigma'}(\Sigma', E') = (\Sigma, E)$.

(ii) If $\mathscr{A}' \in \mathrm{Alg}(\Sigma', E')$, then the *restriction* of $\mathscr{A}'$ to Σ is obtained by deleting all constants, functions, predicates in $\mathscr{A}'$ corresponding to symbols in $\Sigma' - \Sigma$. We write $\rho_{\Sigma}^{\Sigma'}(\mathscr{A}') = \mathscr{A}$ for this restriction. $\mathscr{A}$ is also called a *reduct* of $\mathscr{A}'$, and $\mathscr{A}'$ is called an *expansion* of $\mathscr{A}$.

We will also write $\mathscr{A} \leq \mathscr{A}'$.

(iii) Let $X \subseteq A$. Then $\mathscr{A}_X$ is the expansion of $\mathscr{A}$ obtained by adding the $a \in X$ as designated constants. Instead of $\mathscr{A}_A$ we write $\underset{\sim}{\mathscr{A}}$.

Example: For $\mathscr{A}$ as in Definition 2.1. (iii), $\underset{\sim}{\mathscr{A}} = (\mathbb{N}, 0, 1, 2, 3, \ldots, s, p, k)$. (So in $L(\Sigma_{\underset{\sim}{\mathscr{A}}})$ one can refer to all elements of A by name.)

2.4.1. Remark. Note that if $\mathscr{A}' \geq \mathscr{A}$, then $(\Sigma_{\mathscr{A}'}, E_{\mathscr{A}'}) \trianglerighteq (\Sigma_{\mathscr{A}}, E_{\mathscr{A}})$.

2.5. Definition. Let $\mathscr{A}, \mathscr{B} \in \mathrm{Alg}(\Sigma)$. Then:

(i) $\mathscr{A} \equiv \mathscr{B}$ ($\mathscr{A}, \mathscr{B}$ are *elementary equivalent*) iff $E_{\mathscr{A}} = E_{\mathscr{B}}$.

(ii) Let $A \subseteq B$. Then $\mathscr{A} \preccurlyeq \mathscr{B}$ iff $\underset{\sim}{\mathscr{A}} \equiv \mathscr{B}_A$.

($\mathscr{A}$ is an elementary sub-algebra of $\mathscr{B}$, or $\mathscr{B}$ is an *elementary extension of* $\mathscr{A}$.)

2.5.1. Remark. Note that $\mathscr{A} \preccurlyeq \mathscr{B} \Rightarrow \mathscr{A} \equiv \mathscr{B}$.

2.5.2. Proposition. $\mathscr{A} \preccurlyeq \mathscr{B} \Leftrightarrow \mathscr{B}_A \models E_{\underset{\sim}{\mathscr{A}}}$.

Proof. For the proof, see [24, p. 74]. □

In the sequel we will mostly deal with conservative refinements ($\trianglerighteq$). They have the pleasant property that two refinements $(\Sigma_i, E_i) \trianglerighteq (\Sigma, E)$ $(i = 1, 2)$ can be joined to a refinement $(\Sigma_1 \cup \Sigma_2, E_1 \cup E_2) \trianglerighteq (\Sigma, E)$, provided the obviously necessary requirement that $\Sigma_1 \cap \Sigma_2 = \Sigma$ is satisfied. This is a (strong) form of Robinson's Consistency Theorem (RCT). The version we will need is slightly stronger than the usual statement of RCT. For that reason we include part of the proof. We start with the very useful Joint Consistency Theorem (JCT); for the (hard) proof we refer to [24, p. 79]. From JCT the remaining theorems in this section easily follow. In [21] another order of presentation is followed.

2.6. Joint Consistency Theorem (Craig–Robinson). *Let* (Σ, E) *and* (Σ', E') *be specifications. Then* $E \cup E'$ *is inconsistent iff there is a closed assertion* $p \in L(\Sigma_1 \cap \Sigma_2)$ *such that* $E \vdash p$ *and* $E' \vdash \neg p$.

2.6.1. Corollary (Craig Interpolation Lemma). *Let p and q be closed assertions such that* $\vdash p \rightarrow q$. *Then there is a closed assertion r such that*

(i) $\vdash p \rightarrow r$ *and* $\vdash r \rightarrow q$,

(ii) *every nonlogical symbol occurring in r, occurs in both p and q.*

Proof. Clearly the specification $\{p, \neg q\}$ is inconsistent: $\{p\} \cap \{\neg q\} \vdash p, p \rightarrow q, q, \neg q$, **false**. Hence by Theorem 2.6 there exists a closed assertion $r \in L(\{p, \neg q\})$

such that $\{p\} \vdash r$ and $\{\neg q\} \vdash \neg r$. By the Deduction Theorem it follows that $\vdash p \to r$ and $\vdash \neg q \to \neg r$. □

2.6.2. Corollary (Robinson's Consistency Theorem) (see Fig. 3). *Let* $(\Sigma_i, E_i) \trianglerighteq (\Sigma_0, E_0)$, $i = 1, 2$, *such that* $\Sigma_1 \cap \Sigma_2 = \Sigma_0$. *Then*

(i) $E_1 \cup E_2$ *is consistent, moreover*

(ii) $(\Sigma_1 \cup \Sigma_2, E_1 \cup E_2) \trianglerighteq (\Sigma_0, E_0)$, *and even*

(iii) $(\Sigma_1 \cup \Sigma_2, E_1 \cup E_2) \trianglerighteq (\Sigma_i, E_i)$ $(i = 1, 2)$.

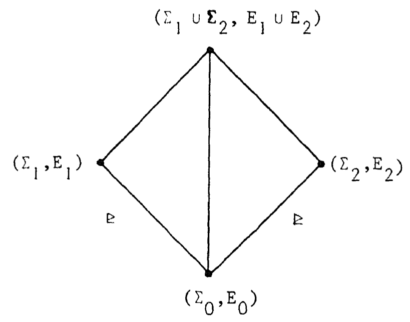


Fig. 3.

Proof. Part (i) immediately follows from (ii), which follows by transitivity of $\trianglerighteq$ from (iii).

Ad (iii): Suppose $E_1 \cup E_2 \vdash p$ for a closed assertion $p \in L(\Sigma_i)$.

Therefore, $\{e_1, e_2\} \vdash p$ for some closed assertions $e_i \in L(\Sigma_i)$, $i = 1, 2$, such that $E_i \vdash e_i$. By the Deduction Theorem:

$$\vdash e_2 \to (e_1 \to p).$$

By Craig's Interpolation Lemma 2.6.1:

$$\vdash e_2 \to r \tag{$\star$}$$

and

$$\vdash r \to (e_1 \to p) \tag{$\star\star$}$$

for some $r \in L(\Sigma_1 \cap \Sigma_2) = L(\Sigma_0)$. By ($\star$), we have $E_2 \vdash r$. Hence $E_0 \vdash r$, since $(\Sigma_2, E_2) \trianglerighteq (\Sigma_0, E_0)$. So, by ($\star\star$), $E_0 \vdash e_1 \to p$. Therefore $E_1 \vdash p$; and this proves $(\Sigma_1 \cup \Sigma_2, E_1 \cup E_2) \trianglerighteq (\Sigma_1, E_1)$. Likewise for (Σ_2, E_2). □

Next, we will give a characterization of the conservativity of refinements. For many purposes, however, the following criterion for conservativity is sufficient.

2.7. Definition. Let (Σ', E') be a refinement such that every $\mathscr{A} \in \mathrm{Alg}(\Sigma, E)$ can be expanded to an $\mathscr{A}' \in \mathrm{Alg}(\Sigma', E')$. Then this refinement is called *simple* (see Fig. 4).

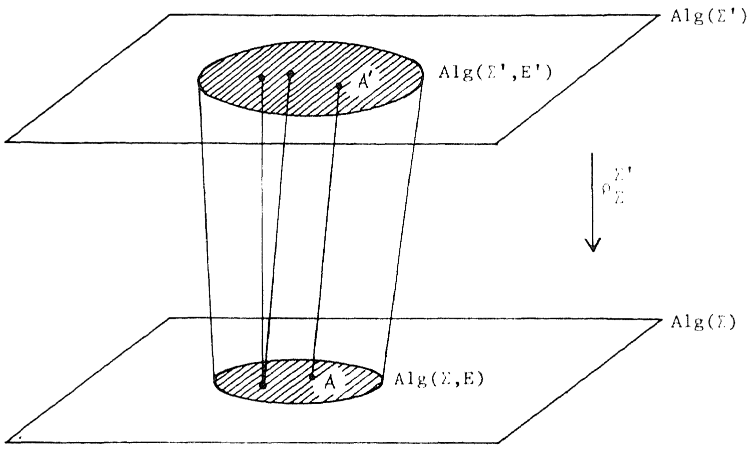


Fig. 4.

2.7.1. Proposition (Criterion for conservativity). *Simple refinements are conservative.*

Proof. Suppose (Σ', E') is a simple refinement of (Σ, E), i.e., $\forall \mathcal{A} \in \mathrm{Alg}(\Sigma, E)$ $\exists \mathcal{A}' \in \mathrm{Alg}(\Sigma', E')$ $\mathcal{A}' \geq \mathcal{A}$. Let $E \nvdash p$ for some closed assertion p. Then by Gödel's Completeness Theorem 1.1.1, $\mathcal{A} \nvDash p$ for some $\mathcal{A} \in \mathrm{Alg}(\Sigma, E)$. So there is an $\mathcal{A}' \in \mathrm{Alg}(\Sigma', E')$ such that $\mathcal{A}' \geq \mathcal{A}$. Hence $\mathcal{A}' \vDash \neg p$; reasoning backwards we have $E' \nvdash p$. $\square$

In general, the situation is more complicated. If $(\Sigma', E') \trianglerighteq (\Sigma, E)$, it may be the case that some $\mathcal{A} \in \mathrm{Alg}(\Sigma, E)$ cannot be expanded to an $\mathcal{A}' \in \mathrm{Alg}(\Sigma', E')$. So we may 'lose' models when taking a refinement. However, such a 'lost' model $\mathcal{A}$ is always an elementary substructure of (and hence elementary equivalent to) an $\mathcal{A}'$ which is not 'lost' (see also Theorem 2.7.3 below).

2.7.2. Example (Shoenfield [24, p. 96]). Let Σ' contain the constant symbols $c_0, c_1, c_2, \ldots$ and let $E' = \{c_i \neq c_j \mid i \neq j\}$. Let (Σ, E) be obtained by omitting c_0 and let $\mathcal{A}$ be $(\mathbb{N} - \{0\}, 1, 2, 3, \ldots)$. Then $\mathcal{A}$ cannot be expanded to an $\mathcal{A}' \in \mathrm{Alg}(\Sigma', E')$, since there is no 'room' for (an interpretation of) c_0.

2.7.3. Theorem (Characterization of conservativity) (see Fig. 5). *Let* $(\Sigma', E') \geq (\Sigma, E)$. *Then the following statements are equivalent*:

(i) $(\Sigma', E') \trianglerighteq (\Sigma, E)$.

(ii) $\forall \mathcal{A} \in \mathrm{Alg}(\Sigma, E)\ \exists \mathcal{A}' \in \mathrm{Alg}(\Sigma, E), \mathcal{A}'' \in \mathrm{Alg}(\Sigma', E')$ *such that* $\mathcal{A} \preccurlyeq \mathcal{A}' \prec \mathcal{A}''$.

(iii) $E' \cup E_{\mathcal{A}}$ *is consistent for all* $\mathcal{A} \in \mathrm{Alg}(\Sigma, E)$.

(iv) $E' \cup E_{\mathcal{A}}$ *is consistent for all* $\mathcal{A} \in \mathrm{Alg}(\Sigma, E)$.

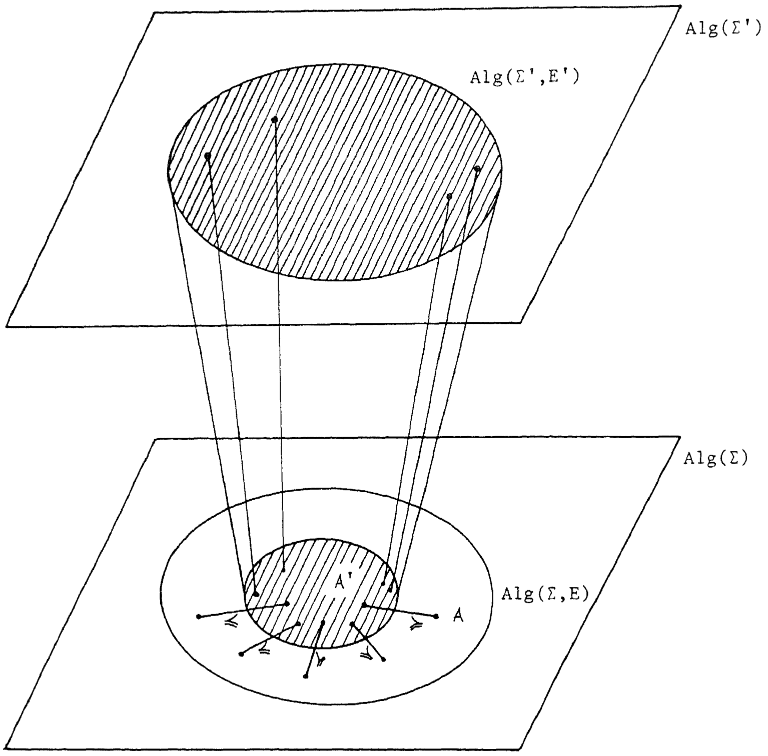


Fig. 5.

Proof. (ii) $\Rightarrow$ (i): Suppose $E \nvdash p$, $p \in L(\Sigma)$. Then $\mathcal{A} \not\models p$ for some $\mathcal{A} \in \mathrm{Alg}(\Sigma, E)$. Now there are $\mathcal{A}' \in \mathrm{Alg}(\Sigma, E)$ and $\mathcal{A}'' \in \mathrm{Alg}(\Sigma', E')$ such that $\mathcal{A} \preccurlyeq \mathcal{A}' \leq \mathcal{A}''$. By Remark 2.5.1, $\mathcal{A} \equiv \mathcal{A}'$. Hence also $\mathcal{A}' \models \neg p$. Therefore, $\mathcal{A}'' \models \neg p$; so $E' \nvdash p$.

(i) $\Rightarrow$ (iii): Let $(\Sigma', E') \unrhd (\Sigma, E)$ and suppose that, for some $\mathcal{A} \in \mathrm{Alg}(\Sigma, E)$, $E' \cup E_{\mathcal{A}}$ is inconsistent. By Theorem 2.6 there is a closed assertion $p \in L(\Sigma' \cap \Sigma_{\mathcal{A}}) = L(\Sigma)$ such that $E' \vdash p$ and $E_{\mathcal{A}} \vdash \neg p$. By conservativity, $E \vdash p$. Hence $\mathcal{A} \models p$; a contradiction with $E_{\mathcal{A}} \vdash \neg p$, because $E_{\mathcal{A}} \vdash \neg p \Leftrightarrow \mathcal{A} \models \neg p \Leftrightarrow \mathcal{A} \models \neg p$.

(iii) $\Rightarrow$ (ii): Suppose $E' \cup E_{\mathcal{A}}$ is consistent. Then there is a $\mathcal{B}''$ such that $\mathcal{B}'' \models E' \cup E_{\mathcal{A}}$. Let $\mathcal{B}'$ be the reduct of $\mathcal{B}''$ to the signature Σ', and let $\mathcal{B}$ be the reduct of $\mathcal{B}''$ to Σ. Then $\mathcal{B}_A \models E_{\mathcal{A}}$, so, by Proposition 2.5.2, $\mathcal{A} \preccurlyeq \mathcal{B}$; and trivially $\mathcal{B} \leq \mathcal{B}'$.

(iii) $\Rightarrow$ (iv): Trivial.

(iv) $\Rightarrow$ (iii): Suppose $E' \cup E_{\mathcal{A}}$ is inconsistent. Then, by Theorem 2.6, $E' \vdash p$ and $E_{\mathcal{A}} \vdash \neg p$ for some $p \in L(\Sigma' \cap \Sigma_{\mathcal{A}}) = L(\Sigma)$. Now $E_{\mathcal{A}} \vdash \neg p \Rightarrow E_{\mathcal{A}} \vdash \neg p$, since $E_{\mathcal{A}}$ is complete. Hence $E' \cup E_{\mathcal{A}}$ is inconsistent. $\square$

2.7.3.1. Example. Let $\mathcal{N} = (\mathbb{N}, 0, 1, +, \times)$ and let $\mathcal{N}^*$ be some non-standard model of arithmetic, so $\mathcal{N}^* \equiv \mathcal{N}$. Then $(\Sigma_{\mathcal{N}^*}, E_{\mathcal{N}^*} \unrhd (\Sigma_{\mathcal{N}}, E_{\mathcal{N}})$.

Proof: $E_{\mathcal{N}^*} \cup E_{\mathcal{A}}$ is consistent for every $\mathcal{A} \in \mathrm{Alg}(\Sigma_{\mathcal{N}}, E_{\mathcal{N}})$ (i.e., every $\mathcal{A}$ such that $\mathcal{A} \equiv \mathcal{N}$) because $E_{\mathcal{A}} = E_{\mathcal{N}} \subseteq E_{\mathcal{N}^*}$. (Note that this refinement is not simple).

3. Definability

We now turn to a special kind of simple conservative refinement (the definitional refinement), collect some material about definability, and use this to prove that '+' is not definable in the algebra $(\mathbb{N}, 0, S, P)$ which will play an important role later on.

3.1. Definition. Let $\Delta \subseteq \Sigma$ and consider (Σ, E). An n-ary predicate symbol $\pi \in \Sigma - \Delta$ is *definable in terms of Δ in E*, if there is an assertion $p \in L(\Delta)$ such that

$$E \vdash \pi(x_1, \ldots, x_n) \leftrightarrow p$$

(where $x_1, \ldots, x_n$ are distinct variables). An n-ary function symbol $\phi \in \Sigma - \Delta$ is definable in terms of Δ in E if there is an assertion $p \in L(\Delta)$ such that

$$E \vdash \phi(x_1, \ldots, x_n) = y \leftrightarrow p$$

(where $x_1, \ldots, x_n, y$ are distinct variables).

3.2. Definition. $(\Sigma', E') \trianglerighteq_d (\Sigma, E)$, in words: (Σ', E') is a definitional refinement of (Σ, E), if $(\Sigma', E') \trianglerighteq (\Sigma, E)$ and every symbol $\in \Sigma' - \Sigma$ is definable in terms of Σ in E'.

3.3. Theorem (Padoa's method). *Let $(\Sigma \cup \{\tau\}, E)$ be a specification where $\tau \notin \Sigma$. Then τ is not definable in terms of Σ in E, if there are two models $\mathcal{A}$, $\mathcal{B} \in \mathrm{Alg}(\Sigma \cup \{\tau\}, E)$ such that* $A = B$ and $\sigma^{\mathcal{A}} = \sigma^{\mathcal{B}}$ *for every nonlogical symbol $\sigma \in \Sigma$, but $\tau^{\mathcal{A}} \neq \tau^{\mathcal{B}}$.*

Proof. Let τ be a predicate symbol. (The proof for function symbols, including the constant symbols which can be considered as '0-ary' function symbols, is similar.) Suppose $\mathcal{A}$, $\mathcal{B}$ exist as given in the theorem, and suppose that τ is definable in terms of Σ in E. That is,

$$E \vdash \tau(\boldsymbol{x}) \leftrightarrow p,$$

for some assertion $p \in L(\Sigma)$. Then for $\boldsymbol{a} \in A$ we have

$$\boldsymbol{a} \in \tau^{\mathcal{A}} \Leftrightarrow \mathcal{A} \models p[\boldsymbol{a}] \Leftrightarrow \mathcal{B} \models p[\boldsymbol{a}] \Leftrightarrow \boldsymbol{a} \in \tau^{\mathcal{B}}$$

(where the middle equivalence follows since $p \in L(\Sigma)$ and $\mathcal{A}$, $\mathcal{B}$ have the same interpretation for every symbol in Σ). Hence $\tau^{\mathcal{A}} = \tau^{\mathcal{B}}$, contradiction. $\square$

3.3.1. Remark. (i) A much stronger theorem results when, in Theorem 3.3, 'if' is replaced by 'iff', namely Beth's Definability Theorem (BDT).

(ii) Write $(\Sigma', E') \geq^1 (\Sigma, E)$ iff $\Sigma' - \Sigma$ is a singleton. Then the version of BDT as indicated in (i) can be paraphrased as $(\Sigma', E') \trianglerighteq^1_{\mathrm{d}} (\Sigma, E) \Leftrightarrow$ the mapping $\rho^{\Sigma'}_{\Sigma} : \mathrm{Alg}(\Sigma', E')$ is injective.

A slightly stronger version of BDT as, e.g., in [24, p. 81] says the same for $\trianglerighteq_{\mathrm{d}}$ instead of $\trianglerighteq^1_{\mathrm{d}}$.

Noting further that $\trianglerighteq_{\mathrm{d}}$ implies $\trianglerighteq_{\mathrm{s}}$, we have the following model theoretic characterization of definitional refinements:

$$(\Sigma', E') \trianglerighteq_{\mathrm{d}} (\Sigma, E) \Leftrightarrow$$

$$\Leftrightarrow \rho^{\Sigma'}_{\Sigma} : \mathrm{Alg}(\Sigma', E') \to \mathrm{Alg}(\Sigma, E) \text{ is injective}$$

$$\Leftrightarrow \rho^{\Sigma'}_{\Sigma} : \mathrm{Alg}(\Sigma', E') \to \mathrm{Alg}(\Sigma, E) \text{ is bijective.}$$

3.3.2. Application. In the sequel we will need the following.

Fact. Let $\mathscr{A} = (\mathbb{N}, 0, S, P)$. Then the function $+$ is not definable in $\mathscr{A}$.

Proof (by Padoa's method). (For another proof, using elimination of quantifiers, see Section 8.) Suppose $+$ is definable in $\mathscr{A}$; i.e., for some assertion r we have

$$\mathscr{A} \models r[a, b, c] \Leftrightarrow a + b = c.$$

Now let $\mathscr{A}' = (\mathbb{N}, 0, S, P, +)$, so

$$\mathscr{A}' \models r(x, y, z) \leftrightarrow x + y = z.$$

Hence

$$E_{\mathscr{A}'} \vdash r(x, y, z) \leftrightarrow x + y = z,$$

so the symbol $+$ is definable in terms of $\Sigma_{\mathscr{A}}$ in $E_{\mathscr{A}'}$.

To show that this is contradictory, we use Padoa's method (Theorem 3.3): We will try to find $\mathcal{N}_1, \mathcal{N}_2, \in \mathrm{Alg}(\Sigma_{\mathscr{A}'}, E_{\mathscr{A}'})$ such that $N_1 = N_2$, $\sigma^{\mathcal{N}_1} = \sigma^{\mathcal{N}_2}$ for all $\sigma \neq +$, but $+^{\mathcal{N}_1} \neq +^{\mathcal{N}_2}$. Two such models are readily obtained; we have to take 'non-standard' models:

$$\mathcal{N}_i = (\mathbb{N} \times \{0\}) \cup (\mathbb{Z} \times \mathbb{N}^+), O_0, S, P, +_i) \quad (i = 1, 2),$$

where $\mathbb{N}^+ = \mathbb{N} - \{0\}$, and where we write a_b instead of (a, b). Further, $S(n_m) = (n+1)_m$, $P(n+1)_m = n_m$, $P(O_0) = O_0$ and $n_m +_i n'_{m'} = (n + n')_{i(m+m')}$ $(i = 1, 2)$.

(Intuitively; the n_0 are the standard numbers; there are nonstandard numbers divided in copies of $\mathbb{Z}$, indexed by positive integers. The point is that these indices are so to speak indiscernible for the specification in question, so there is considerable liberty in defining '+' on the non-standard part.)

3.3.3 Example. *Some reducts of arithmetic.* In the schema given by Fig. 6 most of the above concepts are illustrated. Upward lines denote conservative refinements (of the theory of the structure in question); the 'clusters' of structures are equivalence

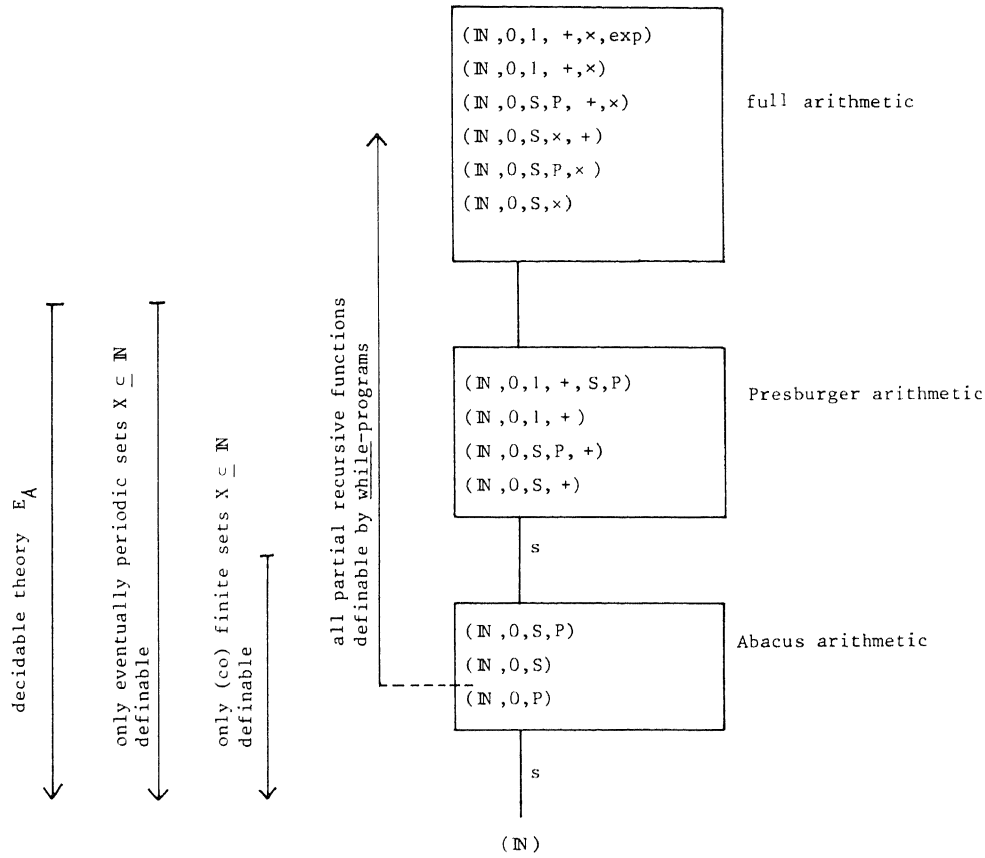


Fig. 6.

classes w.r.t. the equivalence generated by $\trianglerighteq_d$. Simple refinements are indicated with 's'. The most remarkable facts here are the definability of exponentiation from $0, 1, +, \times$, which is well known; and less well known, the definablity of $+$ in terms of $0, S, \times$, by the following:

$$i+j=k \Leftrightarrow (i'k'')'(j'k'')' = ((i'j')'(k''k''))',$$

where $x' = Sx$, $x'' = S(Sx)$ (see [11, p. 219]).

4. Program inclusions

We will now introduce the various notions of the inclusion $\sqsubseteq$ between statements $S_1, S_2 \in \mathcal{WP}(\Sigma)$ that we will study, and prove some elementary facts about them.

4.1. Definition. Let $S \in \mathcal{WP}(\Sigma)$ and $\mathcal{A} = (A, \ldots) \in \mathrm{Alg}(\Sigma, E)$. Let S contain the variables $x_1, \ldots, x_n$ $(n \geq 1)$. Then $S^{\mathcal{A}} : A^n \to A^n$ is the partial function determined by S, i.e.,

$$S^{\mathscr{A}}(a_1,\ldots,a_n)=\begin{cases}(b_1,\ldots,b_n) & \text{if } S \text{ converges with input}\\ & (a_1,\ldots,a_n) \text{ and yields } (b_1,\ldots,b_n),\\ \text{undefined} & \text{otherwise.}\end{cases}$$

4.1.1. Remark. The restriction to functions $f: A^n \to A^n$ is not essential. Instead of, e.g., $f(x_1, x_2, x_3) = x_1 \cdot x_2$ one may use $f'(x_1, x_2, x_3) = (x_1 \cdot x_2, 0, 0)$.

4.2. Definition (Semantical inclusion). Let $S_1, S_2 \in \mathscr{WP}(\Sigma)$. Then

(i) $\quad \mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2 \Leftrightarrow S_1^{\mathscr{A}} \subseteq S_2^{\mathscr{A}} \quad$ for all $\mathscr{A} \in \mathrm{Alg}(\Sigma, E)$.

This inclusion is said to be semantical. Instead of the left-hand side we will also use the notation $S_1 \sqsubseteq_{\mathrm{Alg}(\Sigma,E)} S_2$.

(ii) *Semantical equivalence* w.r.t. (Σ, E) is defined by

$$\mathrm{Alg}(\Sigma, E) \models S_1 \equiv S_2 \Leftrightarrow \mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2 \;\&\; \mathrm{Alg}(\Sigma, E) \models S_2 \sqsubseteq S_1.$$

4.3. Definition (Proof-theoretical inclusion)

(i) $\quad S_1 \sqsubseteq_{\mathrm{HL}(\Sigma,E)} S_2\quad$ iff, for all $p, q \in L(\Sigma)$,

$$\mathrm{HL}(\Sigma, E) \vdash \{p\}\, S_2\, \{q\} \Rightarrow \mathrm{HL}(\Sigma, E) \vdash \{p\}\, S_1\, \{q\}.$$

(Note the direction of the implication. Intuitively: S_1 is less defined than S_2 so $\{p\}\, S_1\, \{q\}$ is more often trivially true.)

(ii) $S_1 \equiv_{\mathrm{HL}(\Sigma,E)} S_2$ is the corresponding equivalence.

4.4. Definition (Derivable inclusion)

(i) $\quad \mathrm{HL}(\Sigma, E) \vdash S_1 \sqsubseteq S_2 \Leftrightarrow \forall (\Sigma', E') \trianglerighteq (\Sigma, E) S_1 \sqsubseteq_{\mathrm{HL}(\Sigma',E')} S_2$.

(The terminology '*derivable*' and the choice of the notation '$\vdash$' is motivated by the sequel: it will be proved that derivable inclusion w.r.t. (Σ, E) is semi-decidable in E.) As before we define $\mathrm{HL}(\Sigma, E) \vdash S_1 \equiv S_2$ *derivable equivalence* w.r.t. (Σ, E).

(ii) $\quad \mathrm{HL}(\Sigma, E) \vdash_{\mathrm{f}} S_1 \sqsubseteq S_2 \Leftrightarrow \forall (\Sigma', E') \trianglerighteq_{\mathrm{f}} (\Sigma, E) S_1 \sqsubseteq_{\mathrm{HL}(\Sigma',E')} S_2$.

4.5. Definition (Forced inclusion)

$$\mathrm{HL}(\Sigma, E) \Vdash S_1 \sqsubseteq S_2 \Leftrightarrow \exists (\Sigma', E') \trianglerighteq (\Sigma, E)\;\; \mathrm{HL}(\Sigma', E') \vdash S_1 \sqsubseteq S_2.$$

As before, *forced equivalence* w.r.t. (Σ, E) is defined.

4.6. Definition. The inclusion $S_1 \sqsubseteq S_2$ is *cofinal*, iff

$$\forall (\Sigma', E') \trianglerighteq (\Sigma, E)\;\; \exists (\Sigma'', E'') \trianglerighteq (\Sigma', E')\;\; S_2 \sqsubseteq_{\mathrm{HL}(\Sigma'',E'')} S_2.$$

It is clear that all inclusions ($\sqsubseteq$) defined above are partial orders and that all equivalences ($\equiv$) are equivalence relations, except for forced and cofinal inclusion resp. equivalence. For the last case, 'cofinal', we will eventually prove that 'cofinal $\Leftrightarrow$ semantical', hence cofinal inclusion is indeed transitive. We will now prove that also forced inclusion is transitive—hence it is a partial order and forced equivalence is an equivalence relation indeed. First we need a simple proposition about renaming of symbols.

4.7. Definition. $(\Sigma_1, E_1) \cong (\Sigma_2, E_2)$ ((Σ_1, E_1) and (Σ_2, E_2) are *isomorphic* specifications) if (Σ_1, E_1) can be obtained from (Σ_2, E_2) by renaming some of the nonlogical symbols; distinct symbols must be replaced by distinct symbols.

4.7.1. Remark. So Robinsons Consistency Theorem 2.6.2 says (see Fig. 7) that if $(\Sigma_i, E_i) \unrhd (\Sigma, E)$, $i = 1, 2$, then for some variant $(\Sigma_2', R_2') \cong (\Sigma_2, E_2)$ such that $(\Sigma_2', E_2') \unrhd (\Sigma, E)$ there exists a $(\Sigma_3, E_3) \unrhd (\Sigma_1, E_1), (\Sigma_2', E_2')$.

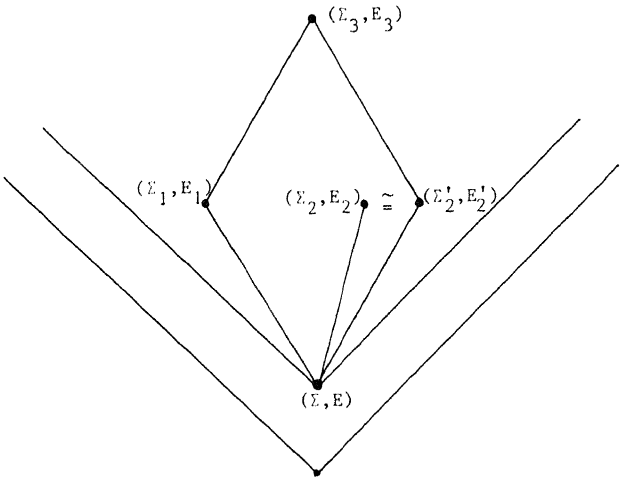


Fig. 7.

4.7.2. Proposition. *Let* $S_1, S_2 \in \mathcal{WP}(\Sigma)$. *Suppose*

$$(\Sigma', E'), (\Sigma'', E'') \unrhd (\Sigma, E), \quad (\Sigma', E') \cong (\Sigma'', E'') \quad \text{and} \quad \Sigma' \cap \Sigma'' = \Sigma.$$

Then

(i) $\quad S_1 \sqsubseteq_{\mathrm{HL}(\Sigma', E')} S_2 \Leftrightarrow S_1 \sqsubseteq_{\mathrm{HL}(\Sigma'', E'')} S_2$,

(ii) $\quad \mathrm{HL}(\Sigma', E') \vdash S_1 \sqsubseteq S_2 \Leftrightarrow \mathrm{HL}(\Sigma'', E'') \vdash S_1 \sqsubseteq S_2$.

Proof. (i) routine; (ii) at once from (i). $\square$

4.8. Proposition. *Let* $S_1, S_2, S_3 \in \mathscr{WP}(\Sigma)$. *Then*

$$\mathrm{HL}(\Sigma, E) \Vdash S_1 \sqsubseteq S_2 \; \& \; \mathrm{HL}(\Sigma, E) \Vdash S_2 \sqsubseteq S_3 \Rightarrow \mathrm{HL}(\Sigma, E) \Vdash S_1 \sqsubseteq S_3.$$

Proof. The assumptions are

$$\exists (\Sigma_i', E_i') \trianglerighteq (\Sigma, E) \; \forall (\Sigma_i'', E_i'') \trianglerighteq (\Sigma_i', E_i') \; S_i \sqsubseteq_{\mathrm{HL}(\Sigma'', E'')} S_{i+1} \quad (i = 1, 2)$$

(see Fig. 8).

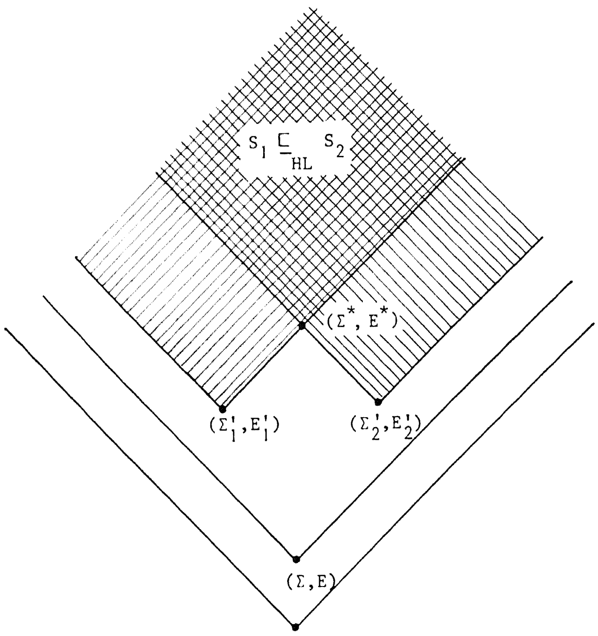


Fig. 8.

Now consider such a (Σ_i', E_i'), $i = 1, 2$. By Proposition 4.7.2 we may suppose that $\Sigma_1' \cap \Sigma_2' = \Sigma$. Now by Robinsons Consistency Theorem 2.6.2, $(\Sigma^*, E^*) = (\Sigma_1' \cup \Sigma_2', E_1' \cup E_2') \trianglerighteq (\Sigma, E)$. Also, by transitivity of $\sqsubseteq_{\mathrm{HL}}$, in the 'upper cone' of (Σ^*, E^*) we have $S_1 \sqsubseteq_{\mathrm{HL}} S_2$. Hence $\mathrm{HL}(\Sigma, E) \Vdash S_1 \sqsubseteq S_3$. $\square$

Another corollary of Robinson's Consistency Theorem (RCT) 2.6.2 is the following.

4.9. Proposition. *Forced inclusion implies cofinal inclusion.*

Proof. Suppose $\mathrm{HL}(\Sigma, E) \Vdash S_1 \sqsubseteq S_2$, i.e.,

$$\exists (\Sigma', E') \trianglerighteq (\Sigma, E) \quad \forall (\Sigma'', E'') \trianglerighteq (\Sigma', E') \quad S_1 \sqsubseteq_{\mathrm{HL}(\Sigma'', E'')} S_2 \tag{1}$$

We have to prove the following (see Fig. 9):

$$\forall(\Sigma_1', E_1') \trianglerighteq (\Sigma, E)\ \exists(\Sigma_1'', E_1'') \trianglerighteq (\Sigma_1', E_1')\ \ S_1 \sqsubseteq_{\mathrm{HL}(\Sigma'', E'')} S_2. \tag{2}$$

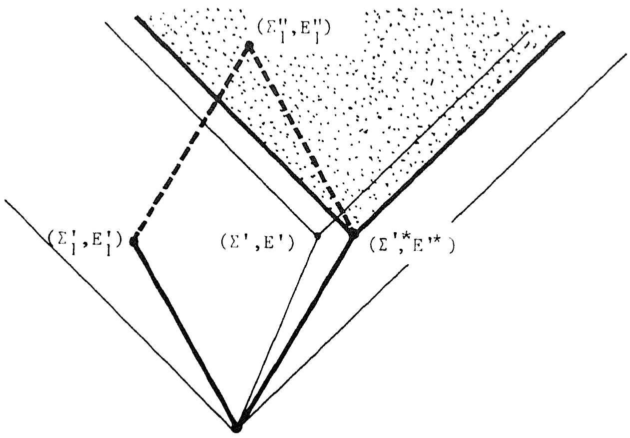


Fig. 9.

Take (Σ', E') as in (1), and consider a (Σ_1', E_1') as in (2). By Proposition 4.7.2(ii) we can 'shift' (Σ', E') to an isomorphic variant (Σ'^*, E'^*) such that $\Sigma'^* \cap \Sigma' = \Sigma$, and still having the property that $S_1 \sqsubseteq_{\mathrm{HL}} S_2$ in all refinements.

Then take (Σ_1'', E_1'') in (2) as the union of (Σ_1', E_1') and (Σ'^*, E'^*); by RCT 2.6.2 this is possible. □

4.9.1 Remark. For $\geq$ instead of $\trianglerighteq$ the above proposition fails. E.g., take

$S_1 = x := 0$

$S_2 =$ **if** $0 > 1$ **then** $x := 0$ **else** $x := 1$ **fi**.

Let $\Sigma = \{0, 1, <\}$, E is the theory of partial order, $E_1 = E \cup \{0 < 1\}$ and $E_2 = E \cup \{0 > 1\}$. Then $\mathrm{HL}(\Sigma, E_2)$ '$\vdash$' $S_1 \equiv S_2$, hence $\mathrm{HL}(\Sigma, E)$ '$\Vdash$' $S_1 \equiv S_2$. However, for all $(\Sigma', E') \geq (\Sigma, E_1)$, $S_1 \not\sqsupseteq_{\mathrm{HL}(\Sigma', E')} S_2$.

4.10. Remark. All inclusions introduced above, except semantical inclusion, were obtained by quantification over the 'basic' proof-theoretical inclusion $\sqsubseteq_{\mathrm{HL}}$. This suggests looking at all inclusions of the following general form:

$$\begin{aligned} S_1 \sqsubseteq^{\forall\exists\forall\cdots\exists}_{\mathrm{HL}(\Sigma,E)} S_2 \Leftrightarrow\ & \forall(\Sigma_1, E_1) \trianglerighteq (\Sigma, E)\ \exists(\Sigma_2, E_2) \trianglerighteq (\Sigma_1, E_1) \\ & \forall(\Sigma_3, E_3) \trianglerighteq (\Sigma_2, E_2) \cdots \exists(\Sigma_{2n}, E_{2n}) \trianglerighteq (\Sigma_{2n-1}, E_{2n-1}) \\ & S_1 \sqsubseteq_{\mathrm{HL}(\Sigma_{2n}, E_{2n})} S_2 \end{aligned}$$

and likewise $S_1 \sqsubseteq^{\forall\exists\forall\cdots\forall}_{\mathrm{HL}(\Sigma,E)} S_2$, and the dual notions obtained by interchanging $\exists, \forall$.

(Note that only alternating strings of quantifiers are interesting, since obviously $--\forall\forall-- = --\forall--$ and likewise for $\exists$.) So derivable inclusion w.r.t. (Σ, E) is $\sqsubseteq^{\forall}_{\mathrm{HL}(\Sigma,E)}$, forced inclusion is $\sqsubseteq^{\exists\forall}_{\mathrm{HL}(\Sigma,E)}$, and cofinal inclusion is $\sqsubseteq^{\forall\exists}_{\mathrm{HL}(\Sigma,E)}$. (In the sequel we will also consider 'inclusion in some refinement': $\sqsubseteq^{\exists}_{\mathrm{HL}(\Sigma,E)}$.)

Now between these generalized inclusions there are a priori the following implications (see Fig. 10 where an implication is downward). (Only the quantifiers of $\sqsubseteq^{\forall\exists--}_{\mathrm{HL}(\Sigma,E)}$ are mentioned.)

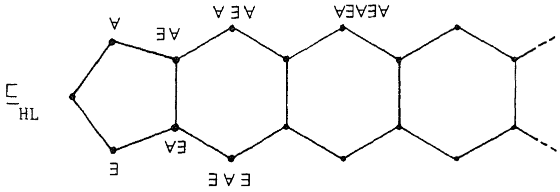


Fig. 10.

However, this hierarchy of inclusions 'collapses' because

(i) $\sqsubseteq^{\exists\forall}_{\mathrm{HL}(\Sigma,E)} = \sqsubseteq^{\forall\exists\forall}_{\mathrm{HL}(\Sigma,E)}$,

(ii) $\sqsubseteq^{\forall\exists}_{\mathrm{HL}(\Sigma,E)} = \sqsubseteq^{\exists\forall\exists}_{\mathrm{HL}(\Sigma,E)}$.

To see the nontrivial direction of (i), note that it was already proved in Proposition 4.9. By a similar argument (ii) also follows.

Now $\exists\forall\exists\forall = \exists\exists\forall = \exists\forall$, $\forall\exists\forall\exists\forall = \forall\exists\forall = \exists\forall$, etc. Hence the only inclusions are those displayed in Fig. 11.

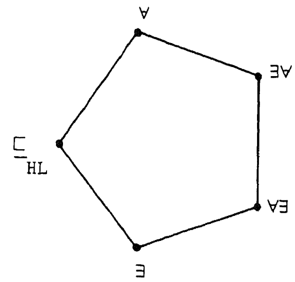


Fig. 11.

(*Remark*: We did not prove that $\sqsubseteq^{\exists}_{\mathrm{HL}(\Sigma,E)}$ is a partial order. *Question*: Is it?).

4.11. Remark. All inclusions that are defined above exhibit the desirable property of staying valid in a context: let $S_1, S_2 \in \mathscr{W}\mathscr{P}(\Sigma)$ and let $C[\ \ ]$ be a *context statement* (also in Σ), i.e., a statement with a 'hole'. Then

$$S_1 \sqsubseteq S_2 \Leftrightarrow \forall C[\ \ ]\ C[S_1] \sqsubseteq C[S_2].$$

The proof follows in a straightforward manner by observing that

$$\forall p, q \in L(\Sigma)\ \ \mathrm{HL}(\Sigma, E) \vdash \{p\}\, S_2\, \{q\} \Rightarrow \mathrm{HL}(\Sigma, E) \vdash \{p\}\, S_1\, \{q\}$$

implies

$$\forall p, q \in L(\Sigma)\ \ \mathrm{HL}(\Sigma, E) \vdash \{p\}\, C[S_2]\, \{q\} \Rightarrow \mathrm{HL}(\Sigma, E) \vdash \{p\}\, C[S_1]\, \{q\}.$$

4.12. Remark (Invariances). For a better insight in what happens inside the 'cone of refinements', we will investigate whether the notions

(1) $\mathrm{Alg}(\Sigma, E) \models p \qquad E \vdash p,$

(2) $\mathrm{Alg}(\Sigma, E) \models \{p\}\, S\, \{q\}; \quad \mathrm{HL}(\Sigma, E) \vdash \{p\}\, S\, \{q\},$

(3) $\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2; \qquad S_1 \sqsubseteq_{\mathrm{HL}(\Sigma, E)} S_2$

are invariant under 'shifting (Σ, E) upward or downward'.

Ad (1). Upward and downward invariant (i.e., $\forall (\Sigma', E') \trianglerighteq (\Sigma, E)$ $(\mathrm{Alg}(\Sigma, E) \models p \Leftrightarrow \mathrm{Alg}(\Sigma', E') \models p)$); this follows simply from Gödels Completeness Theorem 1.1.1 and the definition of conservativity.

Ad (2). Here the situation is already somewhat more complicated: $\mathrm{Alg}(\ ,\ ) \models \{p\}\, S\, \{q\}$ is upward and downward invariant (see Proposition 4.13). However, for $\mathrm{HL}(\ ,\ ) \vdash \{p\}\, S\, \{q\}$ we only have the (trivial) upward invariance, i.e.,

$$\forall (\Sigma', E') \trianglerighteq (\Sigma, E)\ \ \mathrm{HL}(\Sigma, E) \vdash \{p\}\, S\, \{q\} \Rightarrow \mathrm{HL}(\Sigma', E') \vdash \{p\}\, S\, \{q\}.$$

That here '$\Leftarrow$' does not hold, is because an invariant needed for the proof of $\vdash \{p\}\, S\, \{q\}$ may be available in (Σ', E') but not yet in (Σ, E).

Ad (3). Again the semantical notion, $\mathrm{Alg}(\ ,\ ) \models S_1 \sqsubseteq S_2$, is invariant in both directions. For 'upward' this is trivial; for 'downward' certainly not (see Lemma 4.14).

Finally, $S_1 \sqsubseteq_{\mathrm{HL}(,)} S_2$ is neither upward, nor downward invariant. One can even show that it may happen that $S_1 \sqsubseteq_{\mathrm{HL}(,)} S_2$ is alternatingly true and false while following some upward path $(\Sigma_0, E_0) \trianglelefteq (\Sigma_1, E_1) \trianglelefteq \cdots$.

4.13. Proposition. *Let* $(\Sigma', E') \trianglelefteq (\Sigma, E)$, $p, q \in L(\Sigma)$ *and* $S \in \mathscr{WP}(\Sigma)$. *Then* $\mathrm{Alg}(\Sigma, E) \models \{p\}\, S\, \{q\} \Leftrightarrow \mathrm{Alg}(\Sigma', E') \models \{p\}\, S\, \{q\}$.

Proof. ($\Rightarrow$). Trivial.

($\Leftarrow$). To prove the reverse, we use Theorem 2.7.3, which says that for every $\mathscr{A} \in \mathrm{Alg}(\Sigma, E)$ there is an $\mathscr{A}' \in \mathrm{Alg}(\Sigma, E)$ and an $\mathscr{A}'' \in \mathrm{Alg}(\Sigma', E')$ such that $\mathscr{A} \preccurlyeq \mathscr{A}' \leq \mathscr{A}''$. By Remark 2.5.1 we have $\mathscr{A} \equiv \mathscr{A}'$. Now the result follows by the following lemma from [7]: "Let $\mathscr{A} \equiv \mathscr{B}$. Then $\mathscr{A} \models \{p\}\, S\, \{q\} \Leftrightarrow \mathscr{B} \models \{p\}\, S\, \{q\}$". □

4.14. Lemma. *Let* $(\Sigma', E') \trianglerighteq (\Sigma, E)$. *Then, for all* $S_1, S_2 \in \mathscr{WP}(\Sigma)$,

$$\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2 \Leftrightarrow \mathrm{Alg}(\Sigma', E') \models S_1 \sqsubseteq S_2.$$

Proof. ($\Rightarrow$) is easy: take $\mathscr{A}' \in \mathrm{Alg}(\Sigma', E')$. Then $\rho_{\Sigma}^{\Sigma'}(\mathscr{A}') = \mathscr{A} \in \mathrm{Alg}(\Sigma, E)$. So $\mathscr{A} \models S_1 \sqsubseteq S_2$. But then trivially also $\mathscr{A}' \models S_1 \sqsubseteq S_2$, since the extra structure on $\mathscr{A}'$ does not play a role.

($\Leftarrow$). The proof of the reverse follows by contraposition: Take $\mathscr{A} \in \mathrm{Alg}(\Sigma, E)$ such that $\mathscr{A} \not\models S_1 \sqsubseteq S_2$. Then there are $\boldsymbol{a} = a_1, \ldots, a_n \in A$ and $\boldsymbol{b} = b_1, \ldots, b_n \in A$ such that, par abus de language:

$$\mathscr{A} \models S_1(\underline{\boldsymbol{a}}) = \underline{\boldsymbol{b}} \quad \text{and} \quad \mathscr{A} \not\models S_2(\underline{\boldsymbol{a}}) = \underline{\boldsymbol{b}}.$$

More precisely, for some n and for all m:

$$\mathscr{A} \models \phi_n(\underline{\boldsymbol{a}}, \underline{\boldsymbol{b}}) \wedge \neg\psi_m(\underline{\boldsymbol{a}}, \underline{\boldsymbol{b}}),$$

where

$$\phi_n(\underline{\boldsymbol{a}}, \underline{\boldsymbol{b}}) = \mathrm{Comp}_{S_1,n}(\underline{\boldsymbol{a}}) = \underline{\boldsymbol{b}} \quad \text{and} \quad \psi_m(\underline{\boldsymbol{a}}, \underline{\boldsymbol{b}}) = \neg\mathrm{Comp}_{S_2,m}(\underline{\boldsymbol{a}}) = \underline{\boldsymbol{b}}.$$

Let Γ be the set of assertions $\{\phi_n(\underline{\boldsymbol{a}}, \underline{\boldsymbol{b}})\} \cup \{\psi_m(\boldsymbol{a}, \boldsymbol{b}) \mid m \in \mathbb{N}\}$.

Claim. For some $\mathscr{B}$, $\mathscr{B} \models E' \cup \Gamma$. So $\mathscr{B} \not\models S_1 \sqsubseteq S_2$, hence $\mathrm{Alg}(\Sigma', E') \not\models S_1 \sqsubseteq S_2$ and we are through.

Proof of the claim. Suppose there is no such $\mathscr{B}$, i.e., $E' \cup \Gamma$ is inconsistent. Then for some finite $\Delta \subseteq \Gamma$ we have that $E' \cup \Delta$ is already inconsistent. Say $\Delta = \{\phi_n, \neg\psi_0, \ldots, \neg\Psi_{k-1}\}$. So $E' \vdash \neg(\phi_n \wedge \bigwedge_{i<k} \Psi_i)$, hence

$$E' \vdash \neg\exists \boldsymbol{x}, \boldsymbol{y}\, (\phi_n(\boldsymbol{x}, \boldsymbol{y}) \wedge \bigwedge_{i<k} \psi_i(\boldsymbol{x}, \boldsymbol{y})).$$

By the conservativity of E' over E we can replace E' here by E. However, this contradicts the fact that

$$\mathscr{A} \models \exists \boldsymbol{x}, \boldsymbol{y}\, (\phi_n(\boldsymbol{x}, \boldsymbol{y}) \wedge \bigwedge_{i<k} \psi(\boldsymbol{x}, \boldsymbol{y})). \qquad \Box$$

5. Prototype proofs

Let us abbreviate the implication

$$\mathrm{HL}(\Sigma', E') \vdash \{p\}\, S_2\, \{q\} \Rightarrow \mathrm{HL}(\Sigma', E') \vdash \{p\}\, S_1\, \{q\}$$

by $\Phi(\Sigma', E', p, q)$. So, by definition, $\mathrm{HL}(\Sigma, E) \vdash S_1 \sqsubseteq S_2$ is equivalent to

$$\phi(\Sigma', E', p, q) \quad \textit{for all } (\Sigma', E') \trianglerighteq (\Sigma, E) \textit{ and all } p, q \in L(\Sigma').$$

Now it turns out that among all these $\Phi(\Sigma', E', p, q)$ there is a 'generic' one, $\Phi(\Sigma^0, E^0, r(\boldsymbol{x}), r'(\boldsymbol{x}))$. I.e.,

$$\Phi(\Sigma^0, E^0, r(\boldsymbol{x}), r'(\boldsymbol{x})) \Leftrightarrow$$

$$\Leftrightarrow \forall (\Sigma', E') \trianglerighteq (\Sigma, E) \;\; \forall p, q \in L(\Sigma') \;\; \Phi(\Sigma', E', p, q).$$

The situation is even further simplified, since the generic implication has an antecedent $\mathrm{HL}(\Sigma^0, E^0) \vdash \{r(\boldsymbol{x})\}\, S_2\, \{r'(\boldsymbol{x})\}$ which is always true. This reduces checking whether $\mathrm{HL}(\Sigma, E) \vdash S_1 \sqsubseteq S_2$ or not, to checking whether $\mathrm{HL}(\Sigma^0, E^0) \vdash \{r(\boldsymbol{x})\}\, S_1\, \{r'(\boldsymbol{x})\}$, which is semi-decidable. (Hence our choice of the notation $\vdash$ in $HL(\Sigma, E) \vdash S_1 \sqsubseteq S_2$.)

Finding this generic implication is based on the observation that every proof $\mathrm{HL}(\Sigma', E') \vdash \{p\}\, S\, \{q\}$ can be viewed as an instantiation of a *prototype proof* $\pi(S)$. In order to define this concept, we need an efficient notation for proofs of asserted programs. One method is to consider a proof as a proof tree; a second way is to consider a proof as a flow-diagram with assertions written at the cut-points. We will use a more workable *linear* notation of proofs which will be introduced now. First we will define the concept 'interpolated statement' which can be viewed as the flow-diagram corresponding to the statement plus some assertions written at some cutpoints.

5.1. Definition. The class $\mathrm{IStat}(\Sigma)$, with typical elements $S^*, S_1^*, S^{**}, \ldots,$ of *interpolated* statements is inductively defined by

$$S^* ::= S \mid \{p\}S^* \mid S^*\{p\} \mid \textbf{if } b \textbf{ then } S_1^* \textbf{ else } S_2^* \textbf{ fi} \mid \textbf{while } b \textbf{ do } S^* \textbf{ od}.$$

Here $S \in \mathscr{WP}(\Sigma)$. So the class of interpolated statements contains next to the usual statements also asserted statements and statements interlaced with assertions in an arbitrary way; but it contains also *proofs* of asserted statements. These will be singled out by means of the following extended proof rules.

5.2. Definition. By means of the following axioms and extended proof rules we can derive proofs of asserted statements:

(1) *Assignment axiom scheme*:

$$\{p[t/x]\}\; x := t\, \{p\}$$

(2) *Extended composition rule*:

$$\frac{\{p\}\, S_1^*\, \{r\} \quad \{r\}\, S_2^*\, \{q\}}{\{p\}\, S_1^*\, \{r\}\, S_2^*\, \{q\}}$$

(3) *Extended conditional rule*:

$$\frac{\{p \wedge b\}\, S_1^*\, \{q\} \quad \{p \wedge \neg b\}\, S_2^*\, \{q\}}{\{p\}\; \textbf{if } b \textbf{ then } \{p \wedge b\}\, S_1^*\, \{q\} \textbf{ else } \{p \wedge \neg b\}\, S_2^* \{q\} \textbf{ fi } \{q\}}$$

(4) *Extended iteration rule*:

$$\frac{\{p \wedge b\}\, S^*\, \{p\}}{\{p\}\; \textbf{while } b \textbf{ do } \{p \wedge b\}\, S^*\, \{p\} \textbf{ od } \{p \wedge \neg b\}}$$

(5) *Extended consequence rule*:

$$\frac{p \to p \quad \{p_1\}\, S^*\, \{q_1\} \quad q_1 \to q}{\{p\} \quad \{p_1\}\, S^*\, \{q_1\} \quad \{q\}}$$

5.3. Definition and notation. (i) Let $\Pr(\Sigma, E)$ be the class of proofs (interpolated statements) which can be derived using this axiom scheme and extended proof rules, such that in rule (5) only implications provable from E are used.

(ii) If $S^* \in \mathrm{IStat}(\Sigma)$, then $\sigma(S^*)$ will denote the underlying statement obtained by erasing all $\{p\}$ in S^*. (So σ can be inductively defined as follows:

$$\sigma(S) = S \quad \text{for } S \in \mathcal{WP}(\Sigma)$$

$$\sigma(S^*\{p\}) = \sigma(\{p\}\, S^*) = \sigma(S^*)$$

$$\sigma(\textbf{if } b \textbf{ then } S_1^* \textbf{ else } S_2^* \textbf{ fi}) = \textbf{if } b \textbf{ then } \sigma(S_1^*) \textbf{ else } \sigma(S_2^*) \textbf{ fi}$$

$$\sigma(\textbf{while } b \textbf{ do } S^* \textbf{ od}) = \textbf{while } b \textbf{ do } \sigma(S^*) \textbf{ od}.)$$

(iii) If $S^* \in \Pr(\Sigma, E)$, then $\kappa(S^*)$ will denote the set of consequences $p \to p'$ used in the derivation of S^*. Note that these consequences can be read of directly from S^*: $\kappa(S^*) = \{p \to p' \mid \{p\}\{p'\} \subseteq S^*\}$. (Here '$\subseteq$' denotes the relation of being contained as a 'subword'.)

(iv) If $S^* \in \Pr(\Sigma, E)$ and $S^* = \{p\}\, S_1^*\, \{q$, then $\mathrm{pre}(S^*) = p$ and $\mathrm{post}(S^*) = q$.

(v) Let $S^* \in \Pr(\Sigma, E)$. Then S^* is called a *reduced* proof, iff it contains no occurrence of a triple $\{p\}\{q\}\{r\}$. (By the transitivity of $\to$, every proof may be supposed to be reduced, up to equivalence.)

5.4. Definition. (1) Two interpolated statements S^*, S^{**} such that $\sigma(S^*) = \sigma(S^{**}) = S$ are called *matching* if at every place the same number of assertions occur in S^*, S^{**}. (*Notation*: $S^* \sim S^{**}$.)

To be precise:

(i) $S \sim S$ for $S \in \mathcal{WP}(\Sigma)$,

(ii) $S^* \sim S^{**} \Rightarrow \{p\}\, S^* \sim \{q\}\, S^{**}$ and $S^*\{p\} \sim S^{**}\{q\}$
for all assertions $p, q \in L(\Sigma)$,

(iii) $S_1^* \sim S_1^{**}, S_2^* \sim S_2^{**} \Rightarrow$
if b **then** S_1^* **else** S_2^* **fi** $\sim$ **if** b **then** S_1^{**} **else** S_2^{**} **fi**,

(iv) $S^* \sim S^{**} \Rightarrow$
while b **also** S^* **od** $\sim$ **while** b **do** S^{**} **od**.

(2) Let $S^* = {-}{-}\{p\}{-}{-}$ be an interpolated statement containing $\{p\}$. Then $S^{**} = {-}{-}\{p\}\{p\}{-}{-}$ is called a *trivial expansion* of S^*.

5.5. Definition. In the following definition we will use a set of n-ary relation symbols $\{r_i \mid i \in \omega\}$. If $S^* \in \mathrm{IStat}$ contains some of these r-symbols, $[S^*]_j$ will be the result of replacing each occurrence of r_i in S^* by $r_{(i,j)}$ where $(\,,):\mathbb{N}^2\to\mathbb{N}$ is the usual bijective pairing function. (This device merely serves to 'refresh' the r-symbols where necessary.)

(i) Let $S\in\mathcal{WP}(\Sigma)$ involve the variables $\boldsymbol{x}\ (=x_1,\ldots,x_n)$. By induction on the structure of S we define $\pi'(S)$ as follows:

(1) $\quad \pi'(x_i := t)=\{r_0(\boldsymbol{x})\,[t/x_i]\}\ x_i := t\,\{r_0(\boldsymbol{x})\}.$

(2) $\quad \pi'(S_1\,;S_2)=[\pi'(S_1)]_0\,[\pi'(S_2)]_1.$

(That is, $\pi'(S_1)$ and $\pi'(S_2)$ are concatenated, without infix. Moreover, the r-symbols in $[\pi'(S_1)]_0$ are made distinct from those in $[\pi'(S_2)]_1$.)

(3) $\quad \pi'(\textbf{if } b \textbf{ then } S_1 \textbf{ else } S_2 \textbf{ fi}) =$

$$=\{r_0(\boldsymbol{x})\}\ \textbf{if}\ b\ \textbf{then}\ \{r_0(\boldsymbol{x})\wedge b\}\,[\pi'(S_1)]_2\,\{r_1(\boldsymbol{x})\}$$
$$\textbf{else}\ \{r_0(\boldsymbol{x})\wedge\neg b\}\,[\pi'(S_2)]_3\,\{r_1(\boldsymbol{x})\}$$
$$\textbf{fi}\ \{r_1(\boldsymbol{x})\}.$$

(4) $\quad \pi'(\textbf{while } b \textbf{ do } S \textbf{ od}) =$

$$=\{r_0(\boldsymbol{x})\}\ \textbf{while}\ b\ \textbf{do}\ \{r_0(\boldsymbol{x})\wedge b\}\ S^*\ \textbf{od}\,\{r_0(\boldsymbol{x})\wedge\neg b\}\,\{r_1(\boldsymbol{x})\}$$

where $S^*=[\pi'(S)]_4$ and $r_0(\boldsymbol{x})=\mathrm{post}(S^*)$.

(ii) Now $\pi(S)=\{r_0(\boldsymbol{x})\}\,[\pi'(S)]_0\,\{r_1(\boldsymbol{x})\}$. $\pi(S)$ is called the *prototype proof* of S.

5.5.1. Example. Let S be $x_1:=0$; $x_2:=1$; **while** $x_2>x_3$ **do if** $x_1=0$ **then** $x_3:=0$ **else** $x_1:=x_2+1$ **fi od**; $x_1:=x_1+x_2$. Then

$\pi(S)=$

$\{r_1(x_1,x_2,x_3)\}$

$\{r_2(0,x_2,x_3)\}$

$x_1:=0$

$\{r_2(x_1,x_2,x_3)\}$

$\{r_3(x_1,1,x_3)\}$

$x_2:=1$

$\{r_3(x_1,x_2,x_3)\}$

$\{r_6(x_1,x_2,x_3)\}$

while $x_2>x_3$ **do**

$\{r_6(x_1,x_2,x_3)\wedge x_2>x_3\}$

$\{r_4(x_1,x_2,x_3)\}$

if $x_1 = 0$ **then**

$\{r_4(x_1, x_2, x_3) \wedge x_1 = 0\}$

$\{r_5(x_1, x_2, 0)$

$x_3 := 0$

$\{r_5(x_1, x_2, x_3)\}$

$\{r_6(x_1, x_2, x_3)\}$

else

$\{r_4(x_1, x_2, x_3) \wedge \neg x_1 = 0\}$

$\{r_7(x_2 + 1, x_2, x_3)\}$

$x_1 := x_2 + 1$

$\{r_7(x_1, x_2, x_3)\}$

$\{r_6(x_1, x_2, x_3)\}$

fi

$\{r_6(x_1, x_2, x_3)\}$

od

$\{r_6(x_1, x_2, x_3) \wedge \neg\ x_2 > x_3\}$

$\{r_8(x_1 + x_2, x_2, x_3)\}$

$x_1 := x_1 + x_2$

$\{r_8(x_1, x_2, x_3)\}$

$\{r_9(x_1, x_2, x_3)\}$

5.5.2. Proposition. *Let r be a 'new' relation symbol occurring in $\pi(S)$. Then r has an occurrence in $\pi(S)$ of the form $\{r(\mathbf{x})\}$, i.e., the arguments are all variables.*

Proof. Evident by inspection of the definition of $\pi(S)$. □

5.6. Definition. Let $S^* \in \mathrm{IStat}(\Sigma)$ contain the n-ary relation symbol r, and let $p = p(x_1, \ldots, x_n) \in L(\Sigma)$. (Note that p may contain other variables than those displayed.)

Then $\phi_r^p(S^*)$ is the result of replacing each $r(t_1, \ldots, t_n)$, occurring in S^*, by $p(t_1, \ldots, t_n)$. Likewise we define $\phi_{r_1,\ldots,r_n}^{p_1,\ldots,p_r}(S^*)$.

5.6.1. Remark. One can think of the prototype proof $\pi(S)$ as an initial object in the category of proofs $\{p\}\, S^*\, \{q\}$ (where $\sigma(S^*) = S$); morphisms between proofs are the substitutions ϕ.

5.7. Lemma. *Let* $S^* \in \mathrm{Pr}(\Sigma, E)$ *be a reduced proof such that* $\sigma(S^*) = S$. *Then* $\phi: \pi(S) \to S^*$ *for some substitution* ϕ *as in Definition* 5.6. (*So every proof is an instance of the prototype proof.*)

Proof. Consider S, S^* as in the claim of the lemma. We may suppose that S^* and $\pi(S)$ are matching; if not, only some trivial expansions (Definition 5.4) of S^* are required.

We will construct by induction on the structure of S a substitution $\phi: \pi(S) \to S^*$.

Case 1. $S = x := t(\boldsymbol{y}, x, \boldsymbol{z})$, where all variables in t are displayed. Now

$$\pi(S) = \{r_1(\boldsymbol{y}, x, \boldsymbol{z})\}\, \{r_2(\boldsymbol{y}, t, \boldsymbol{z})\}\; x := t\, \{r_2(\boldsymbol{y}, x, \boldsymbol{z})\}\, \{r_3(\boldsymbol{y}, x, \boldsymbol{z})\}$$

and

$$S^* = \{p_1\}\, \{p_2[t/x]\}\; x := t\, \{p_2\}\, \{p_3\}.$$

So the substitution will be $\phi: r_i(\boldsymbol{y}, x, \boldsymbol{z}) \mapsto p_i$ $(i = 1, 2, 3)$.

Case 2. $S = S_1 ; S_2$. So $S^* = \{p_0\}\, \{p_1\}\, S_1^*\, \{p_2\}\, S_2^*\, \{p_3\}\, \{p_4\}$.

By induction hypothesis we have substitutions

$$\phi_1: \pi(S_1) \to \{p_1\}\, S_1^*\, \{p_2\}, \qquad \phi_2: \pi(S_2) \to \{p_2\}\, S_2^*\, \{p_3\}.$$

Now

$$\begin{aligned}\pi(S_1 : S_2) &= \{r_0(\boldsymbol{x})\}\; \pi'(S_1)\; \pi'(S_2)\, \{r_1(\boldsymbol{x})\}\\ &= \underline{\{r_0(\boldsymbol{x})\} \cdots \{r_0'(\boldsymbol{x})\}}\,\underline{\{r_1'(\boldsymbol{x})\} \cdots \{r_1(\boldsymbol{x})\}}\end{aligned}$$

where ——— $= \pi(S_1)$ and - - - - $= \pi(S_2)$. From this it is evident how to construct the desired ϕ. (*Remark*: The arity of the new r-symbols in $\pi(S_i)$, $i = 1, 2$, is that of S (i.e., n if S has the variables $x_1, \ldots, x_n$).)

Case 3. $S =$ **if** b **then** S_1 **else** S_2 **fi**. Then $\pi(S)$ and S^* are as follows:

$$\begin{aligned}\pi(S) = \{r_0(\boldsymbol{x})\}\, \{r_1(\boldsymbol{x})\}\ &\mathbf{if}\ b\ \mathbf{then}\ \{r_1(\boldsymbol{x}) \wedge b\}\; \pi'(S_1)\, \{r_2(\boldsymbol{x})\}\\ &\qquad\mathbf{else}\{r_1(\boldsymbol{x}) \wedge \neg b\}\; \pi'(S_2)\, \{r_2(\boldsymbol{x})\}\\ &\mathbf{fi}\, \{r_2(\boldsymbol{x})\}\, \{r_3(\boldsymbol{x})\},\end{aligned}$$

$$\begin{aligned}S^* = \{p_0\}\{p_1\}\ &\mathbf{if}\ b\ \mathbf{then}\, \{p_1 \wedge b\}\; S_1^*\, \{p_2\}\\ &\qquad\mathbf{else}\ \{p_1 \wedge \neg b\}\; S_2^*\, \{p_2\}\\ &\mathbf{fi}\, \{p_2\}\, \{p_3\}.\end{aligned}$$

Again $\phi: r_i(\boldsymbol{x}) \mapsto p_i$ $(i = 0, 1, 2, 3)$; the induction hypothesis takes care of the correspondence between $\pi'(S_i)$ and S_i^* $(i = 1, 2)$.

Case 4. $S = \textbf{while } b \textbf{ do } S' \textbf{ od}$. (In the following '$r_i$' stands for '$r_i(\boldsymbol{x})$'.)

$$\begin{array}{rcccccccc} \pi(S) = & \{r_0\} & \{r_1\} & \textbf{while } b \textbf{ do} & \{r_1 \wedge b\} & \pi'(S') & \textbf{od} & \{r_1 \wedge \neg b\} & \{r_2\} \\ \phi: & \Big\downarrow & \Big\downarrow & & \Big\downarrow & \Big\downarrow{\scriptstyle \text{induction hypothesis}} & & \Big\downarrow & \Big\downarrow \\ S^* = & \{p_0\} & \{p_1\} & \textbf{while } b \textbf{ do} & \{p_1 \wedge b\} & S^* & \textbf{od} & \{p_1 \wedge \neg b\} & \{p_2\} \end{array}$$

Here $r_1 = \text{post}(\pi'(S'))$ and $p_1 = \text{post}(S^*)$. $\square$

In the sequel we will need a simple proof-theoretical fact, stating that derivability in first order predicate logic is invariant under substitutions ϕ (as in Definition 5.6).

5.8. Proposition. *Let (Σ, E) be a specification and $p, q \in L(\Sigma)$. Let ϕ be a substitution of assertions p_i for relation symbols r_i, as in Definition 5.6. (The p_i's are not necessarily in $L(\Sigma)$.) Let $\phi(E) = \{\phi(p') \mid p' \in E\}$. Then*

(i) $E \vdash p \Rightarrow \phi(E) \vdash \phi(p)$,

(ii) $E \vdash p \rightarrow \phi(E) \vdash \phi(p) \rightarrow \phi(q)$.

Proof. (i) A routine induction on the length of the derivation $E \vdash p$.

(ii) follows from (i), noting that $\phi(p \rightarrow q) = \phi(p) \rightarrow \phi(q)$. $\square$

5.9. Proposition. *Let $\Sigma^0 = \Sigma \cup \Sigma_{\pi(S)}$ and $E^0 = E \cup \kappa(\pi(S))$. Then $(\Sigma^0, E^0) \trianglerighteq_{\mathrm{f}} (\Sigma, E)$.*

Proof. Take arbitrary p, q such that $\mathrm{HL}(\Sigma, E) \vdash \{p\}\, S\, \{q\}$. (E.g., take $q = \textbf{true}$.) Let $\{p\}\, S^*\, \{q\} \in \mathrm{Pr}(\Sigma, E)$ be the corresponding proof; we may suppose it matches $\pi(S)$.

Now let $\mathscr{A} \in \mathrm{Alg}(\Sigma, E)$, so by soundness of HL we have $\mathscr{A} \models \{p\}\, S\, \{q\}$. Further, it is not hard to see that the $r_i(\boldsymbol{x})$ can be interpreted in $\mathscr{A}$ just like the matching assertions in $\{p\}\, S^*\, \{q\}$.

Hence every $\mathscr{A} \in \mathrm{Alg}(\Sigma, E)$ can be expanded to an $\mathscr{A}^0 \in \mathrm{Alg}(\Sigma^0, E^0)$. So, by the conservativity criterium (Proposition 2.7.1), we have $(\Sigma^0, E^0) \trianglerighteq (\Sigma, E)$. The finiteness is obvious. $\square$

5.10. Lemma. *Let $\Sigma^0 = \Sigma \cup \Sigma_{\pi(S_2)}$, $E^0 = E \cup \kappa(\pi(S_2))$ and let $r(\boldsymbol{x})$, $r'(\boldsymbol{x})$ be respectively the assertions at the head and at the tail of $\pi(S_2)$.*

Then the following statements are equivalent:

(i) $\mathrm{HL}(\Sigma, E) \vdash S_1 \sqsubseteq S_2$,

(ii) $\mathrm{HL}(\Sigma, E) \vdash_{\mathrm{f}} S_1 \sqsubseteq S_2$

(iii) $\mathrm{HL}(\Sigma^0, E^0) \vdash \{r(\boldsymbol{x})\}\ S_2\,\{r'(\boldsymbol{x})\} \Rightarrow \mathrm{HL}(\Sigma^0, E^0) \vdash \{r(\boldsymbol{x})\}\ S_1\,\{r'(\boldsymbol{x})\}$

(iv) $\mathrm{HL}(\Sigma^0, E^0) \vdash \{r(\boldsymbol{x})\}\ S_1\,\{r'(\boldsymbol{x})\}$.

Proof. (i) $\Rightarrow$ (ii) is trivial, (ii) $\Rightarrow$ (iii) follows from Proposition 5.9, and (iii) $\Rightarrow$ (iv) follows because it is obvious from the construction that $\mathrm{HL}(\Sigma^0, E) \vdash \{r(\boldsymbol{x})\}\ S_2\,\{r'(\boldsymbol{x})\}$. It remains to prove that (iv) $\Rightarrow$ (i).

Assume (iv): let $\{r_0(\boldsymbol{x})\}\ S_1^*\ \{r_1(\boldsymbol{x})\} \in \mathrm{Pr}(\Sigma^0, E^0)$ be the corresponding proof. Further, suppose for some $(\Sigma', E') \trianglerighteq (\Sigma, E)$, $p, q \in L(\Sigma')$ that we have $\mathrm{HL}(\Sigma', E') \vdash \{p\}\ S_2\,\{q\}$. Let $\{p\}\ S_2^*\ \{q\} \in \mathrm{Pr}(\Sigma', E')$ be the corresponding proof, which we may suppose matching with $\pi(S_2)$. By Lemma 5.7, $\{p\}\ S_2^*\ \{q\}$ is an instance of $\pi(S_2)$ via some substitution ϕ.

Now consider $\phi(\{r_0(\boldsymbol{x})\}\ S_1^*\ \{r_1(\boldsymbol{x}\}) = \{p\}\ \phi(S_1^*)\ \{q\}$. From the construction and by Proposition 5.8 it follows that this is a proof in $\mathrm{Pr}(\Sigma', E')$. Hence $\mathrm{HL}(\Sigma', E') \vdash \{p\}\ S_1\,\{q\}$. □

5.11. Theorem. $\mathrm{HL}(\Sigma, E) \vdash S_1 \sqsubseteq S_2$ *and* $\mathrm{HL}(\Sigma, E) \vdash S_1 \equiv S_2$, *as predicates of* S_1, S_2, *are semi-decidable in* E.

Proof. This follows immediately by noting that (Σ^0, E^0) can effectively be computed from S_2, given (Σ, E), and using the equivalence (i) $\Leftrightarrow$ (iv) in Lemma 5.10. □

6. Completions

In Section 7 we will need the possibility of taking, for given (Σ, E), a refinement $(\Sigma', E') \trianglerighteq (\Sigma, E)$ which is *logically complete* (see Definition 1.2.2). Also we will use a refinement $(\Sigma'', E'') \trianglerighteq (\Sigma, E)$ which *has an SP-calculus* (see Definition 6.3). The concepts and theorems thereabout, used below, are from Bergstra and Tucker [9, 10] and Bergstra and Terlouw [6]. There, however, the following restriction is made: E must have only infinite models. Since we want to develop the present theory in full generality (also for, e.g., $E = \emptyset$), we will extend the above mentioned results by some 'formal' constructions which do not require the restriction on E, and which are made possible by the concept of a prototype proof $\pi(S)$. The disadvantage is that in this way we will need an infinite signature extension $\Sigma' \geq \Sigma$, but for our purpose that is no objection. (*Question*: Given a specification (Σ, E) such that E has finite models, is there a logical complete $(\Sigma \cup \Delta, E') \trianglerighteq (\Sigma, E)$ where Δ is *finite*?)

6.1. Theorem. *For every* (Σ, E) *there is a* $(\Sigma', E') \trianglerighteq (\Sigma, E)$ *such that* (Σ', E') *is logically complete.*

Proof. The proof is by a construction of length ω^2. The first ω steps are as follows.

Enumerate $\mathcal{WP}(\Sigma)$ as $\{S_n \mid n \in \mathbb{N}\}$ and let $\{(p_n, q_n) \mid n \in \mathbb{N}\}$ be an enumeration of the pairs of assertions $\in L(\Sigma)$. Now consider the sequence of asserted programs $\alpha_n = \{p_{(n)_0}\}\, S_{(n)_1}\, \{q_{(n)_0}\}$ where $(\ )_0, (\ )_1$ are the projections corresponding to the well-known bijection $(\ ,\ ): \mathbb{N}^2 \to \mathbb{N}$. Note that every $\{p\}\, S\, \{q\}$ occurs in this sequence.

Now we define by induction on n the specification (Σ_n, E_n).

Basis: $(\Sigma_0, E_0) = (\Sigma, E)$.

Induction step: Let (Σ_n, E_n) be defined, and consider α_{n+1}.

Case 1. $\mathrm{Alg}(\Sigma_n, E_n) \not\models \alpha_{n+1}$. Then $(\Sigma_{n+1}, E_{n+1}) = (\Sigma_n, E_n)$.

Case 2. $\mathrm{Alg}(\Sigma_n, E_n) \models \alpha_{n+1}$. Say the prototype proof $\pi(S_{(n+1)_1})$ has the form $\{r(\boldsymbol{x})\}\, S^*_{(n+1)_1}\, \{r'(\boldsymbol{x})\}$ and let (Σ', E') be the specification corresponding to $\pi(S_{(n+1)_1})$. Then define

$$(\Sigma_{n+1}, E_{n+1}) = (\Sigma_n, E_n) \cup (\Sigma', E' \cup \{p_{(n)_0} \to r(\boldsymbol{x}),\ r'(\boldsymbol{x}) \to q_{(n)_0}\}).$$

(The *r-symbols in* $\pi(S_{(n+1)_1})$ have to be fresh compared to previous r-symbols in (Σ_n, E_n).)

Further, let $(\Sigma_\omega, E_\omega) = \bigcup_{n\in\omega} (\Sigma_n, E_n)$.

Claim 1. $(\Sigma_0, E_0) \trianglelefteq (\Sigma_1, E_1) \trianglelefteq \cdots \trianglelefteq (\Sigma_n, E_n) \trianglelefteq \cdots \trianglelefteq (\Sigma_\omega, E_\omega)$.

Proof of Claim 1. To show that $(\Sigma_n, E_n) \trianglelefteq (\Sigma_{n+1}, E_{n+1})$ for all $n \in \omega$, we use the conservativity criterion of Proposition 2.7.1. Since we know (in Case 2 above) that α_{n+1} is true in every $\mathcal{A} \in \mathrm{Alg}(\Sigma_n, E_n)$, the newly added r-symbols can be interpreted in $\mathcal{A}$; that is, $\mathcal{A}$ can be expanded to an $\mathcal{A}' \in \mathrm{Alg}(\Sigma_{n+1}, E_{n+1})$.

To show that $(\Sigma_n, E_n) \trianglelefteq (\Sigma_\omega, E_\omega)$ for all $n \in \omega$, suppose $E_\omega \vdash p$, for some $p \in L(\Sigma_n)$. Then, for some finite $D \subseteq E_\omega$, $D \vdash p$. Hence, for some $m \geq n$, $E_m \vdash p$. Since $(\Sigma_n, E_n) \trianglelefteq (\Sigma_m, E_m)$ as just shown, $E_n \vdash p$.

Now that $(\Sigma_\omega, E_\omega)$ is constructed, the statements $\in \mathcal{WP}(\Sigma_\omega)$ and assertions $\in L(\Sigma_\omega)$ are again enumerated, and the procedure is repeated to yield $((\Sigma_\omega)_\omega, (E_\omega)_\omega) = (\Sigma_{\omega.2}, E_{\omega.2})$. Likewise $(\Sigma_{\omega.n}, E_{\omega.n})$ is constructed, and we put $(\Sigma', E') = \bigcup_{n\in\omega} (\Sigma_{\omega.n}, E_{\omega.n})$.

Claim 2. $(\Sigma_{\omega.n}, E_{\omega.n}) \trianglelefteq (\Sigma', E')$ for all $n \in \omega$; and (Σ', E') is logically complete.

Proof of Claim 2. The first part is as in the proof of Claim 1. The logical completeness is shown as follows. Let $\mathrm{Alg}(\Sigma', E') \models \{p\}\, S\, \{q\}$, where $\{p\}\, S\, \{q\} \in L(\Sigma')$. Then $\{p\}\, S\, \{q\} \in L(\Sigma_{\omega.n}, E_{\omega.n})$ for some $n \in \omega$, and $\mathrm{Alg}(\Sigma_{\omega.n}, E_{\omega.n}) \models \{p\}\, S\, \{q\}$ follows from Proposition 4.13. (*Alternative argument*: Because no models were 'lost' in the construction, i.e., $\rho(\mathrm{Alg}(\Sigma', E') = \mathrm{Alg}(\Sigma_{\omega.n}, E_{\omega.n})$ for the suitable reduction operator ρ.) Hence $E_{\omega.(n+1)}$ contains $\kappa(\{p\}\, \pi(S)\, \{q\})$, that is, $\mathrm{HL}(\Sigma_{\omega.(n+1)}, E_{\omega.(n+1)}) \vdash \{p\}\, S\, \{q\}$. □

6.2. Corollary. *Let* $\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2$. *Then*

$$\exists (\Sigma', E') \trianglerighteq (\Sigma, E) \quad S_1 \sqsubseteq_{\mathrm{HL}(\Sigma', E')} S_2.$$

Proof. Let (Σ', E') be a logically complete refinement of (Σ, E); by the preceding theorem this exists. By Lemma 4.13 we have

$$\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2 \Leftrightarrow \mathrm{Alg}(\Sigma', E') \models S_1 \sqsubseteq S_2.$$

Now $\mathrm{Alg}(\Sigma', E') \models S_1 \sqsubseteq S_2$ implies

$$\forall p, q \in L(\Sigma')\,(\mathrm{Alg}(\Sigma', E') \models \{p\}\, S_2\, \{q\} \Rightarrow \mathrm{Alg}(\Sigma', E') \models \{p\}\, S_1\, \{q\}).$$

Hence, by logical completeness of (Σ', E'), we have

$$\forall p, q \in L(\Sigma')\,(\mathrm{HL}(\Sigma', E') \vdash \{p\}\, S_2\, \{q\} \Rightarrow \mathrm{HL}(\Sigma', E') \models \{p\}\, S_1\, \{q\}),$$

i.e. $S_1 \sqsubseteq_{\mathrm{HL}(\Sigma',E')} S_2$. $\square$

6.3. Definition. Let (Σ, E) be a specification. We say that (Σ, E) has an SP-calculus (strongest postcondition calculus), if for each $p \in L(\Sigma)$, $S \in \mathcal{WP}(\Sigma)$ there exists an assertion $\mathrm{SP}(p, S) \in L(\Sigma)$ such that

(i) $\mathrm{HL}(\Sigma, E) \vdash \{p\}\, S\, \{\mathrm{SP}(p, S)\}$,

(ii) if $\mathrm{HL}(\Sigma, E) \vdash \{p\}\, S\, \{q\}$, then $(\Sigma, E) \vdash q \rightarrow \mathrm{SP}(p, S)$.

6.4. Theorem. *Let (Σ, E) be a specification without finite models. Then there is a conservative refinement $\mathrm{PA}(\Sigma, E)$ of (Σ, E), called the Peano companion of (Σ, E), which has an* SP-*calculus.*

Proof. For the definition of $\mathrm{PA}(\Sigma, E)$ and the proof that it has an SP-calculus, see [10] and [6]. $\square$

6.4.1. Remark. It is possible to construct a 'formal' companion having an SP-calculus, without the restriction on E, but at the cost of an infinite signature extension. For the sequel we will not need the full strength of an SP-calculus and we will be satisfied with the following proposition.

6.4.2. Proposition. *Let $p, q \in L(\Sigma)$ and $S \in \mathcal{WP}(\Sigma)$.*

(i) *Let $p \leadsto^S q$ abbreviate $\forall(\mathrm{SP}(p, S) \rightarrow q)$, where $\forall$ denotes the universal closure. Then*

$$\mathrm{PA}(\Sigma, E) \vdash \{p \wedge p \leadsto^S q\} S \{q\}$$

(a kind of 'S-modus ponens').

(ii) *Let $p \Rightarrow^S q$ abbreviate $\forall(\bigwedge \kappa(\{p\}\pi(S)\{q\}))$, i.e., the universal closure of the conjunction of the consequences in $\{p\}\, \pi(S)\, \{q\}$. Let $\Sigma' = \Sigma \cup \Sigma_{\pi(S)}$. Then*

$$(\Sigma', \emptyset) \vdash \{p \wedge p \Rightarrow^S q\}\, S\, \{q\}.$$

Proof. (i) Follows at once from the definitions.

(ii) Follows by a tedious but routine verification by induction on S. $\square$

7. Proving program inclusion

We are now in a position to prove one of the main theorems of this paper, viz. the equivalence of semantical and cofinal inclusion. After that we will show how this fact can be exploited to give formal proofs of program inclusion.

7.1. Theorem. *Semantical and cofinal inclusion coincide*; *i.e.*,

$$\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2 \Leftrightarrow \forall(\Sigma', E') \trianglerighteq (\Sigma, E)\ \exists(\Sigma'' E'') \trianglerighteq (\Sigma', E')$$
$$S_1 \sqsubseteq_{\mathrm{HL}(\Sigma'', E'')} S_2.$$

Proof. ($\Rightarrow$). Suppose $\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2$ and consider $(\Sigma', E') \trianglerighteq (\Sigma, E)$. By Theorem 6.1 there is a $(\Sigma'', E'') \trianglerighteq (\Sigma', E')$ which is logically complete. From $\mathrm{Alg}(\Sigma'', E'') \models S_1 \sqsubseteq S_2$ we have

$$\forall p, q \in L(\Sigma'')\ (\mathrm{Alg}(\Sigma'', E'') \models \{p\}\, S_2 \{q\} \Rightarrow \mathrm{Alg}(\Sigma'', E'') \models \{p\}\, S_1\, \{q\}).$$

By the logical completeness we can replace '$\mathrm{Alg}(\Sigma'', E'') \models$' by '$\mathrm{HL}(\Sigma'', E'') \vdash$'. This results in $S_1 \sqsubseteq_{\mathrm{HL}(\Sigma'', E'')} S_2$.

($\Leftarrow$). Let E have no finite models. (The case that E has finite models, can be dealt with analogously, as suggested by Proposition 6.4.2.)

Suppose $\mathrm{Alg}(\Sigma, E) \not\models S_1 \sqsubseteq S_2$. Then also $\mathrm{Alg}(\mathrm{PA}(\Sigma, E)) \not\models S_1 \sqsubseteq S$ by Lemma 4.14. So there is an $\mathscr{A} \in \mathrm{Alg}(\mathrm{PA}(\Sigma, E))$ such that $\mathscr{A} \not\models S_1 \sqsubseteq S_2$. Hence for some $\boldsymbol{a}, \boldsymbol{b} \in A$ we have '$\mathscr{A} \models S_1(\boldsymbol{a}) = \boldsymbol{b}$' but '$\mathscr{A} \models S_2(\boldsymbol{a}) \neq \boldsymbol{b}$', par abus de language. These facts can be properly expressed by

$$\theta = (\boldsymbol{x} = \underline{\boldsymbol{a}} \leadsto^{S_2} \boldsymbol{x} \neq \underline{\boldsymbol{b}}) \wedge \mathrm{Comp}_{n, S_1}(\underline{\boldsymbol{a}}) = \underline{\boldsymbol{b}},$$

for some n (see Computation Lemma 1.1.2). The $\underline{\boldsymbol{a}}$, $\underline{\boldsymbol{b}}$ are new constant symbols. Let $\mathscr{A}' \geq \mathscr{A}$ be the expansion of $\mathscr{A}$ with distinguished elements $\boldsymbol{a}$, $\boldsymbol{b}$, and let (Σ', E') be the conservative refinement of $\mathrm{PA}(\Sigma, E)$ obtained by adding $\boldsymbol{a}$, $\boldsymbol{b}$ to the signature. (By Lemma 2.7.1 this is indeed conservative.) Now

(i) $\mathrm{HL}(\Sigma', E') \vdash \{\theta \wedge \boldsymbol{x} = \underline{\boldsymbol{a}}\}\, S_2\, \{\boldsymbol{x} \neq \underline{\boldsymbol{b}}\}$,

(ii) $\mathrm{HL}(\Sigma', E') \nvdash \{\Theta \wedge \boldsymbol{x} = \underline{\boldsymbol{a}}\}\, S_1\, \{\boldsymbol{x} \neq \underline{\boldsymbol{b}}\}$.

Ad (i). This is Proposition 6.4.2(i).

Ad (ii). $\mathscr{A}' \not\models \{\theta \wedge \boldsymbol{x} = \underline{\boldsymbol{a}}\}\, S_1\, \{\boldsymbol{x} \neq \underline{\boldsymbol{b}}\}$, hence $\mathrm{Alg}(\Sigma', E') \not\models \{\theta \wedge \boldsymbol{x} = \underline{\boldsymbol{a}}\}\, S_1\, \{\boldsymbol{x} \neq \underline{\boldsymbol{b}}\}$. By soundness of HL, (ii) follows.

Finally, we note that (i) also holds in refinements of (Σ', E'), trivially; and the same for (ii) by the downward invariance of $\mathrm{Alg}(,) \models \{p\}\, S\, \{q\}$ (Proposition 4.13). Therefore, $S_1 \sqsubseteq_{(\Sigma'', E'')} S_2$ for all $(\Sigma'', E'') \trianglerighteq (\Sigma', E')$. $\square$

We now know that the schema, given in Fig. 12, holds, and we want to prove that, in general, all implications are displayed in this figure. First we will show in

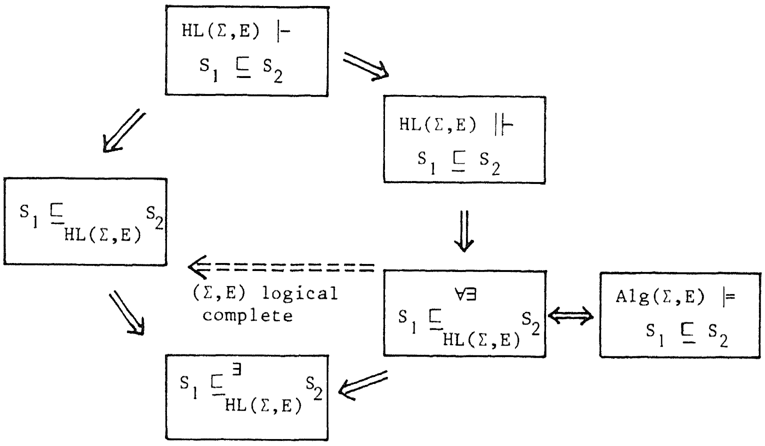


Fig. 12.

Examples 7.2 and 7.3 that $\sqsubseteq_{\mathrm{HL}(\Sigma,E)}$ and $\sqsubseteq_{\mathrm{Alg}(\Sigma,E)}$ are incomparable (see also Fig. 13). Then, in Example 7.4, we show that derivable inclusion is strictly stronger than forced inclusion, in general. (I.e., the proof system corresponding to derivable inclusion proves less inclusions than the one corresponding to forced inclusion.) Further, it will be shown in the next section (Theorem 8.5) that forced inclusion and semantical inclusion are in general not equivalent. In other words, the proof system corresponding to forced inclusion is incomplete.

Finally, at the end of this section (Remark 7.8), we will prove that the 'dashed' implication for logical complete (Σ, E) (see Fig. 12) can in general not be reversed, and we will prove some assertions in the part 'Intuition' of the Introduction.

7.2. Example. Let $\mathcal{A} = (\mathbb{N}, 0, S, P)$, the 'abacus-algebra' as in Section 8, and consider $(\Sigma_{\mathcal{A}}, E_{\mathcal{A}})$. Define

$$S_1 = y := 0;\ S' \text{ where } S' = \textbf{while } x \neq 0 \textbf{ do } y := Sy;\ x := Px \textbf{ od}$$

$$S_2 = y := x;\ x := 0.$$

So $\mathrm{Alg}(\Sigma_{\mathcal{A}}, E_{\mathcal{A}}) \models S_1 \sqsubseteq S_2$. However, $S_1 \not\sqsubseteq_{\mathrm{HL}(\Sigma_{\mathcal{A}}, E_{\mathcal{A}})} S_2$ because

(i) $\mathrm{HL}(\Sigma_{\mathcal{A}}, E_{\mathcal{A}}) \vdash \{x = z\}\ S_2\ \{x = 0 \wedge y = z\}$,

(ii) $\mathrm{HL}(\Sigma_{\mathcal{A}}, E_{\mathcal{A}}) \nvdash \{x = z\}\ S_1\ \{x = 0 \wedge y = z\}$.

Proof of (ii). Suppose not (ii). Then

$$\mathrm{HL}(\Sigma_{\mathcal{A}}, E_{\mathcal{A}}) \vdash \{x = z \wedge y = 0\}\ S'\ \{x = 0 \wedge y = z\}.$$

Hence there must be an invariant $r(x, y, z)$ such that $E_{\mathcal{A}} \vdash \phi_1 \wedge \phi_2 \wedge \phi_3$ where

$$\phi_1 = x = z \wedge y = 0 \rightarrow r(x, y, z),$$

$$\phi_2 = \exists x', y'\ [x' \neq 0 \wedge x = Px' \wedge y = Sy' \wedge r(x', y', z)] \rightarrow r(x, y, z),$$

$$\phi_3 = x = 0 \wedge r(x, y, z) \rightarrow y = z.$$

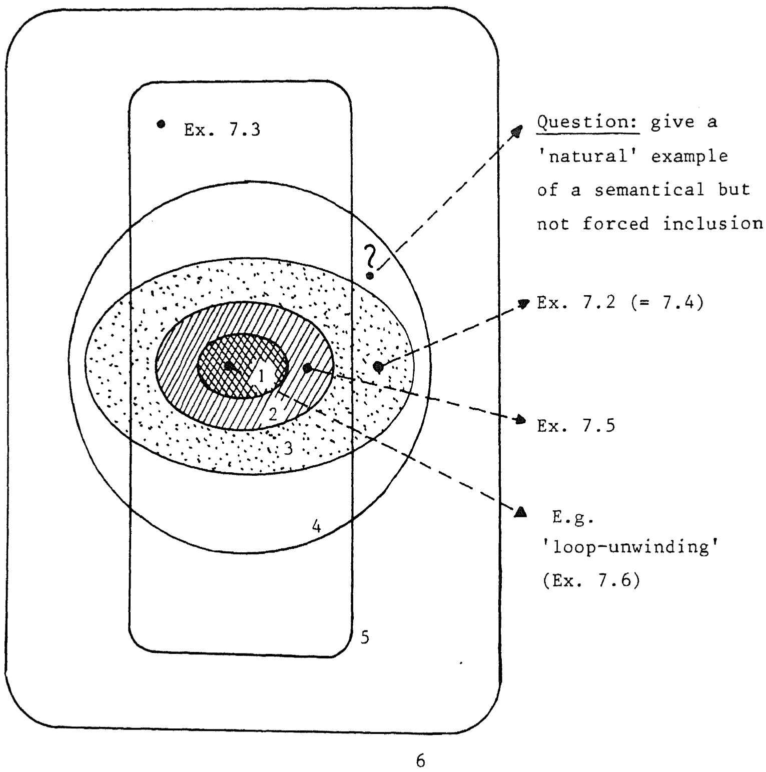


Fig. 13. Venn-diagram of the various notions of inclusion.

1. Logical inclusion (i.e., $\mathrm{HL}(\Sigma, \emptyset) \vdash S_1 \sqsubseteq S_2$, see Examples 7.6 and 7.7).
2. Derivable inclusion.
3. Forced inclusion.
4. Semantical inclusion = cofinal inclusion.
5. Prooftheoretic inclusion.
6. Inclusion in some extension.

Also $\mathscr{A} \models \phi_1 \wedge \phi_2 \wedge \phi_3$. However, a simple proof then shows that $\mathscr{A} \models r(\underline{a}, \underline{b}, \underline{c}) \Leftrightarrow a + b = c$, in contradiction with the non-definability of $+$ in $\mathscr{A}$ (see Remarks 8.3.1 and 3.3.2).

7.3. Example. Let $\mathcal{N} = (\mathbb{N}, 0, S, +, \times)$, Σ the signature of $\mathcal{N}$ and $E = E_{\mathcal{N}}$. Furthermore,

$$S_1 = x := 0;\ \textbf{while}\ x \neq y\ \textbf{do}\ x := x + 1\ \textbf{od}$$

$$S_2 = x := y$$

Then (i) $S_1 \equiv_{\mathrm{HL}(\Sigma, E)} S_2$, but (ii) $S_1 \not\equiv_{\mathrm{Alg}(\Sigma, E)} S_2$.

Proof. (i) HL is relatively complete for $\mathcal{N}$, i.e.,

$$\mathcal{N} \models \{p\}\, S\, \{q\} \Leftrightarrow \mathrm{HL}(\Sigma, E) \vdash \{p\}\, S\, \{q\}.$$

Now $\mathcal{N} \models S_1 \equiv S_2$ implies

$$\forall p, q \;\; \mathcal{N} \models \{p\}\, S_1\, \{q\} \Leftrightarrow \mathcal{N} \models \{p\}\, S_2\, \{q\}$$

or equivalently

$$\forall p, q \;\; \mathrm{HL}(\Sigma, E) \vdash \{p\}\, S_1\, \{q\} \Leftrightarrow \mathrm{HL}(\Sigma, E) \vdash \{p\}\, S_2\, \{q\},$$

i.e., $S_1 \equiv_{\mathrm{HL}(\Sigma,E)} S_2$. Since in our case indeed $\mathcal{N} \models S_1 \equiv S_2$, we have (i).

(ii) However, in a nonstandard model $\mathcal{N}^* \in \mathrm{Alg}(\Sigma, E)$, S_1 will diverge when y is nonstandard. So $\mathcal{N}^* \not\models S_1 \equiv S_2$, hence $\mathrm{Alg}(\Sigma, E) \not\models S_1 \equiv S_2$.

7.4. Example. Back to Example 7.2, which shows moreover that

$$\mathrm{HL}(\Sigma, E) \vdash S_1 \sqsubseteq S_2 \not\Leftarrow \mathrm{HL}(\Sigma, E) \Vdash S_1 \sqsubseteq S_2.$$

From $S_1 \not\sqsubseteq_{\mathrm{HL}(\Sigma_{\mathcal{A}}, E_{\mathcal{A}})} S_2$ it follows trivially that $S_1 \sqsubseteq S_2$ is not derivable. However, for $(\Sigma', E') = (\Sigma_{\mathcal{A}'}, E_{\mathcal{A}'})$ where $\mathcal{A}' = (\mathbb{N}, 0, S, P, +)$ we do have

$$\mathrm{HL}(\Sigma_{\mathcal{A}'}, E_{\mathcal{A}'}) \vdash S_1 \sqsubseteq S_2 \qquad (\bigstar)$$

The proof of $(\bigstar)$ is by the method of prototype proofs, as follows. Consider $\pi(S_2)$, this is given by

$$\{r_0(x, y)\}\, \{r_1(x, x)\}\; y := x\; \{r_1(x, y)\}\, \{r_2(0, y)\} x := 0\, \{r_2(x, y)\}\, \{r_3(x, y)\}.$$

So we have to find a proof of $\{r_0(x, y)\}\, S_1\, \{r_3(x, y)\}$ in the theory

$$E_{\mathcal{A}'} \cup \{r_0(x, y) \to r_1(x, x),\, r_1(x, y) \to r_2(0, y),\, r_2(x, y) \to r_3(x, y)\}.$$

This is indeed possible:

$\{r_0(x, y)\}\, \{r_1(x, x)\}\, \{r_2(0, x)\}\, \{r_3(0, x)\}$

$y := 0$

$\{r_3(0, x) \wedge y = 0\}$

$\{\exists x_0\, [r_3(0, x_0) \wedge x = x_0 \wedge y = 0]\}$

$\{\exists x_0\, [r_3(0, x_0) \wedge x + y = x_0]\}$

while $x \neq 0$ **do**

$\{\exists x_0\, [r_3(0, x_0) \wedge x + y = x_0 \wedge x \neq 0]\}$

$\{\exists x_0\, [r_3(0, x_0) \wedge Px + Sy = x_0 \wedge x \neq 0]\}$

$y := Sy$

$\{\exists x_0\, [r_3(0, x_0) \wedge Px + y = x_0 \wedge x \neq 0]\}$

$x := Px$

$$\{\exists x_0\,[r_3(0, x_0) \wedge x + y = x_0]\}$$

od

$$\{\exists x_0\,[r_3(0, x_0) \wedge x + y = x_0] \wedge x = 0\}$$

$$\{\exists x_0\,[r_3(0, x_0) \wedge y = x_0 \wedge x = 0]\}$$

$$\{r_3(x, y)\}.$$

The above concepts and theorems generalize without any effort (other than notational) to the case of *multi-sorted signatures and algebras.* To substantiate this claim, we give the following example.

7.5. Example. Let Σ be the multi-sorted signature consisting of

domains : NUM, VEC, FUN

constants : $0, 1 \in \mathrm{NUM}$, $\emptyset \in \mathrm{VEC}$

functions : $+: \mathrm{NUM} \times \mathrm{NUM} \to \mathrm{NUM}$

$\cdot: \mathrm{NUM} \times \mathrm{NUM} \to \mathrm{NUM}$

$\mathrm{AP}: \mathrm{VEC} \times \mathrm{NUM} \to \mathrm{VEC}$

$\mathrm{INP}: \mathrm{VEC} \times \mathrm{VEC} \to \mathrm{NUM}$

$\mathrm{ROW}: \mathrm{FUN} \times \mathrm{NUM} \to \mathrm{VEC}$

$\mathrm{EVAL}: \mathrm{FUN} \times \mathrm{NUM} \to \mathrm{NUM}$

variables : $x, y, z \in \mathrm{NUM}$

$X, Y, Z \in \mathrm{VEC}$

$\alpha, \beta \in \mathrm{FUN}$

The specification (Σ, E) we are interested in has the following axioms, describing how the inproduct between two vectors should behave:

$E = \{$Peano + all induction axioms

$$\mathrm{INP}(\emptyset, Z) = \mathrm{INP}(Z, \emptyset) = 0$$

$$\mathrm{INP}(\mathrm{AP}(Z, x), \mathrm{AP}(Z', x')) = \mathrm{INP}(Z, Z') + x \cdot x'$$

$$\mathrm{AP}(Z, x) = \mathrm{AP}(Z', x') \to Z = Z' \wedge x = x'$$

$$\mathrm{ROW}(\alpha, 0) = \emptyset$$

$$\mathrm{ROW}(\alpha, x+1) = \mathrm{AP}(\mathrm{ROW}(\alpha, x), \mathrm{EVAL}(\alpha, x+1))$$

$$\forall x\, \mathrm{EVAL}(\alpha, x) = \mathrm{EVAL}(\beta, x) \to \alpha = \beta\}.$$

Furthermore, let $S_1, S_2 \in \mathscr{WP}(\Sigma)$ be the following programs, both computing the inproduct of two vectors:

$$\begin{aligned} S_1 = {} & A := \emptyset, B := \emptyset;\ z := 0;\ x := 0; \\ & \quad \textbf{while}\ x \neq y\ \textbf{do}\ x := x+1; \\ & \qquad\qquad\qquad z := z + \mathrm{EVAL}(\alpha, x) \cdot \mathrm{EVAL}(\beta, x) \\ & \qquad\qquad \textbf{od}\ x := 0, \end{aligned}$$

$$\begin{aligned} S_2 = {} & A := \mathrm{ROW}(\alpha, y);\ B := \mathrm{ROW}(\beta, y);\ z := \mathrm{INP}(A, B); \\ & x := 0;\ A := \emptyset;\ B := \emptyset. \end{aligned}$$

Now we want to prove that $\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2$. (The reverse does not hold by the presence of nonstandard models in $\mathrm{Alg}(\Sigma, E)$.) (This can be done by proving that $\mathrm{HL}(\Sigma, E) \vdash S_1 \sqsubseteq S_2$, using the method of prototype proofs, as follows. First we write down $\pi(S_2)$:

$$\{r_0(x, y, z, A, B)\}$$

$$\{r_1(x, y, z, \mathrm{ROW}(\alpha, y), B)\}$$

$A := \mathrm{ROW}(\alpha, y)$

$$\{r_1(x, y, z, A, B)\}$$

$$\{r_2(x, y, z, A, \mathrm{ROW}(\beta, y))\}$$

$B := \mathrm{ROW}(\beta, y)$

$$\{r_2(x, y, z, A, B)\}$$

$$\{r_3(x, y, \mathrm{INP}(A, B), A, B)\}$$

$z := \mathrm{INP}(A, B)$

$$\{r_3(x, y, z, A, B)\}$$

$$\{r_4(0, y, z, A, B)\}$$

$x := 0$

$$\{r_4(x, y, z, A, B)\}$$

$$\{r_5(x, y, z, \emptyset, B)\}$$

$A := \emptyset$

$$\{r_5(x, y, z, A, B)\}$$

$$\{r_6(x, y, z, A, \emptyset)\}$$

$B := \emptyset$

$$\{r_6(x, y, z, A, B)\}$$

$$\{r_7(x, y, z, A, B)\}$$

So $\kappa(\pi(S_2))$, the set of consequences used in $\pi(S_2)$, entails the following implications:

$$r_0(x, y, z, A, B) \rightarrow$$

$$r_1(x, y, z, \mathrm{ROW}(\alpha, y), B) \rightarrow$$

$$r_2(x, y, z, \mathrm{ROW}(\alpha, y), \mathrm{ROW}(\beta, y)) \rightarrow$$

$$r_3(x, y, \mathrm{INP}(\mathrm{ROW}(\alpha, y), \mathrm{ROW}(\beta, y)), \mathrm{ROW}(\alpha, y), \mathrm{ROW}(\beta, y)) \rightarrow$$

$$r_4(0, y, \mathrm{INP}(\mathrm{ROW}(\alpha, y), \mathrm{ROW}(\beta, y)), \mathrm{ROW}(\alpha, y), \mathrm{ROW}(\beta, y)) \rightarrow$$

$$r_5(0, y, \mathrm{INP}(\mathrm{ROW}(\alpha, y), \mathrm{ROW}(\beta, y)), \emptyset, \mathrm{ROW}(\beta, y)) \rightarrow$$

$$r_6(0, y, \mathrm{INP}(\mathrm{ROW}(\alpha, y), \mathrm{ROW}(\beta, y)), \emptyset, \emptyset) \rightarrow$$

$$r_7(0, y, \mathrm{INP}(\mathrm{ROW}(\alpha, y), \mathrm{ROW}(\beta, y)), \emptyset, \emptyset).$$

Using these implications together with theory E, we can prove $\{r_0(x, y, z, A, B)\}$ S_1 $\{r_7(x, y, z, A, B)\}$ (and by Lemma 5.10 this proves $\mathrm{HL}(\Sigma, E) \vdash S_1 \sqsubseteq S_2$):

$\{r_0(x, y, z, A, B)\}$

$\{r_7(0, y, \mathrm{INP}(\mathrm{ROW}(\alpha, y), \mathrm{ROW}(\beta, y)), \emptyset, \emptyset)\}$

$A := \emptyset;$

$\{r_7(0, y, \mathrm{INP}(\mathrm{ROW}(\alpha, y), \mathrm{ROW}(\beta, y)), A, \emptyset)\}$

$B := \emptyset;$

$\{r_7(0, y, \mathrm{INP}(\mathrm{ROW}(\alpha, y), \mathrm{ROW}(\beta, y)), A, B)\}$ (abbreviation: r'_7)

$z := 0;$

$\{r'_7 \wedge z = 0\}$

$x := 0;$

$\{r'_7 \wedge z = 0 \wedge x = 0\}$

$\{r'_7 \wedge z = \mathrm{INP}(\mathrm{ROW}(\alpha, x), \mathrm{ROW}(\beta, x))\}$

while $x \neq y$ **do**

$(r'_7 \wedge z = \mathrm{INP}(\mathrm{ROW}(\alpha, x), \mathrm{ROW}(\beta, x)) \wedge x \neq y\}$

$x := x + 1;$

$\{r'_7 \wedge \exists x' (z = \mathrm{INP}(\mathrm{ROW}(\alpha, x'), \mathrm{ROW}(\beta, x')) \wedge x = x' + 1$
$\wedge x' \neq y)\}$

$z := z + \mathrm{EVAL}(\alpha, x) \cdot \mathrm{EVAL}(\beta, x)$

$\{r'_7 \wedge \exists x', z' (z' = \mathrm{INP}(\mathrm{ROW}(\alpha, x'), \mathrm{ROW}(\beta, x') \wedge x = x' + 1$
$\wedge x' \neq y \wedge z = z' + \mathrm{EVAL}(\alpha, x) \cdot \mathrm{EVAL}(\beta, x))\}$

(Now use E)

$$\{r_7' \wedge \exists x' \, (z = \mathrm{INP}(\mathrm{ROW}(\alpha, x'+1), \mathrm{ROW}(\beta, x'+1)) \wedge x = x'+1 \wedge x' \neq y)\}$$

$$\{r_7' \wedge z = \mathrm{INP}(\mathrm{ROW}(\alpha, x), \mathrm{ROW}(\beta, x))\}$$

od

$$\{r_7' \wedge z = \mathrm{INP}(\mathrm{ROW}(\alpha, x), \mathrm{ROW}(\beta, x)) \wedge x = y\}$$

$$\{r_7(0, y, z, A, B)\}$$

$x := 0$

$$\{r_7(x, y, z, A, B)\}.$$

Hence $\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2$.

7.6. Example. Define (as a special case of derivable inclusion) *logical inclusion* of S_1 in S_2 as follows: $\mathrm{HL}(\Sigma, \emptyset) \vdash S_1 \sqsubseteq S_2$. Now the following well-known equivalences are 'logical':

(i) (*Loop-unwinding*)

$S_1 =$ **while** b **do** S **od**; D ($D = x := x$),

$S_2 =$ **if** b **then while** b **do** S **od**; D **else** D.

The proof that $\mathrm{HL}(\Sigma, \emptyset) \vdash S_1 \sqsubseteq S_2$ immediately follows by computing $\pi(S_1)$ and using the thus obtained set of consequences $\kappa(\pi(S_1))$:

$$r_0(x) \to r_1(x),$$

$$r_1(x) \wedge b \to r_2(0), \qquad r_2(x) \to r_1(x),$$

$$r_1(x) \wedge \neg b \to r_3(x),$$

to prove that $\{r_0(x)\}\, S_2\, \{r_3(x)\}$. Likewise for the reverse inclusion.

(ii) Another example of logical inclusion, which is equally simple to verify:

$S_1 =$ **while true do** S **od**, $\qquad S_2$ is arbitrary.

Then $\mathrm{HL}(\Sigma, \emptyset) \vdash S_1 \sqsubseteq S_2$. This example is from [4, p. 93] as well as the next one:

(iii) $S_1 =$ **while** $b_1 \vee b_2$ **do** S **od**

$S_2 =$ **while** b_1 **do** S **od**; **while** b_2 **do** S; **while** b_1 **do** S **od od**.

Here also a simple computation yields the logical equivalence of S_1, S_2.

7.7. Example. Manna [20, p. 251, p. 259] gives several examples of program equivalence which are all 'logical':

(i) $S_1 = x_2 := f(x_1);\ x_2 := g(x_1, x_3) \qquad S_2 = x_2 := g(x_1, x_3)$

(ii) $S_1 = \textbf{while } p(x_2) \textbf{ do } x_1 := g(x_1, x_3) \textbf{ od } D$

$S_2 = \textbf{if } p(x_2) \textbf{ than } \mathrm{DIV} \textbf{ else } D \textbf{ fi}$

Here $\mathrm{DIV} = \textbf{while } x = x \textbf{ do } x := x$, and $D = x := x$

(iii) $S_1 = x := y+1; \textbf{if } x = 1 \textbf{ then } z := 0 \textbf{ else } y := y+1;$

$\textbf{if } y = 1 \textbf{ then } z := 1 \textbf{ else } z := 2 \textbf{ fi fi}$

$S_2 = x := y+1; \textbf{ if } x = 1 \textbf{ then } z := 0 \textbf{ else } y := y+1;$

$z := 2 \textbf{ fi}.$

(Adapted from [20, p. 252]. Note that S_1 contains a useless branch.)

7.8. Remarks. (1) Abbreviate

$$\forall p, q \in L(\Sigma) \ \mathrm{Alg}(\Sigma, E) \models \{p\} S_1 \{q\} \Rightarrow \mathrm{Alg}(\Sigma, E) \models \{p\} S_2 \{q\}$$

by $S_1 \sqsubseteq_{\mathrm{PC}(\Sigma,E)} S_2$ (where PC stands for partial correctness).

Then, for (Σ, E) logically complete, it follows at once from Definition 1.2.2 that $\sqsubseteq_{\mathrm{HL}(\Sigma,E)}$ and $\sqsubseteq_{\mathrm{PC}(\Sigma,E)}$ coincide.

Since $S_1 \sqsubseteq_{\mathrm{Alg}(\Sigma,E)} S_2$ implies $S_1 \sqsubseteq_{\mathrm{PC}(\Sigma,E)} S_2$ (trivially) for all (Σ, E), we have therefore, for logical complete (Σ, E),

$$S_1 \sqsubseteq_{\mathrm{Alg}(\Sigma,E)} S_2 \Rightarrow S_1 \sqsubseteq_{\mathrm{HL}(\Sigma,E)} S_2.$$

The reverse implication does not hold. We give a counterexample:

$S_1 = x := 0, y := 0,$

$S_2 = \textbf{while } x \neq y \textbf{ do } x := x+1 \textbf{ od}; \ x := 0; \ y := 0,$

$(\Sigma, E) = (\Sigma_{\mathcal{N}}, E_{\mathcal{N}})$ where $\mathcal{N} = (\mathbb{N}, 0, 1, +, \times)$.

Now (Σ, E) is logical complete (see [7]) and HL is relatively complete for $\mathcal{N}$ (see [4, Chapter 3]). From the last fact it follows that $S_1 \equiv_{\mathrm{HL}(\Sigma,E)} S_2$. However, due to the presence of nonstandard models in $\mathrm{Alg}(\Sigma, E)$, we have $S_1 \not\equiv_{\mathrm{Alg}(\Sigma,E)} S_2$.

(2) Note that (1) also establishes that (ii) $\not\Rightarrow$ (i) (i.e., $S_1 \sqsubseteq_{\mathrm{PC}(\Sigma,E)} S_2 \not\Rightarrow S_1 \sqsubseteq_{\mathrm{Alg}(\Sigma,E)} S_2$), as claimed in the Introduction. For another counterexample, see [5, Theorem 5.8].

(3) As claimed in the Introduction:

$$\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2 \Leftrightarrow \forall (\Sigma', E') \trianglerighteq (\Sigma, E) S_1 \sqsubseteq_{\mathrm{PC}(\Sigma',E')} S_2.$$

Here $(\Rightarrow)$ is trivial.

Proof of $(\Leftarrow)$: Assume the right-hand side, and suppose $\mathrm{Alg}(\Sigma, E) \not\models S_1 \sqsubseteq S_2$. Then since semantical and cofinal inclusion coincide (Theorem 7.1), we have

$$\exists (\Sigma', E') \trianglerighteq (\Sigma, E) \ \forall (\Sigma'', E'') \trianglerighteq (\Sigma', E') \ S_1 \not\sqsubseteq_{HL(\Sigma'',E'')} S_2.$$

Now consider such a (Σ', E'), and a (Σ'', E'') which is logically complete. Then by the assumption of the right-hand side, $S_1 \sqsubseteq_{PC(\Sigma'', E'')} S_2$; and by logical completeness, $S_1 \sqsubseteq_{HL(\Sigma'', E'')} S_2$; a contradiction.

8. Abacus arithmetic

In this section we will consider our paradigm algebra $\mathcal{A} = (\mathbb{N}, 0, S, P)$. It is useful by the following two well-known facts (already mentioned in Example 3.3.3).

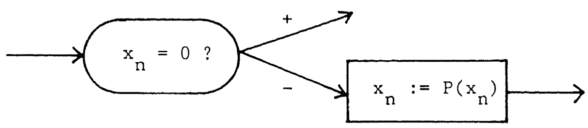
8.1. Proposition. (i) *$E_{\mathcal{A}}$ is a decidable theory, and* (ii) *every partial recursive function can be computed in $\mathcal{A}$ by some $S \in \mathcal{WP}(\Sigma_A)$.*

Using this proposition we will calculate the degrees in the arithmetical hierarchy of the various inclusions $S_1 \sqsubseteq S_2$ (as predicates of S_1, S_2) w.r.t. $(\Sigma_{\mathcal{A}}, E_{\mathcal{A}})$.

For a proof of Proposition 8.1(ii), see, e.g., [11, Chapters 6 and 7], where results from [19] are presented. The proof there uses in fact not **while**-programs, but flow-diagrams composed of only two operations:

assignments: $x_n := S(x_n) \quad (n = 0, 1, 2, \ldots)$

branching operations:



(As pointed out in [19], such a flow-diagram is in fact computing on an *infinite abacus.* Variables as in such a diagram are known as *counters.*) Combined with the equally well-known fact that for every flow-diagram there is an equivalent **while**-program (see, e.g., [19]) we have Proposition 8.1(ii).

For the sake of completeness, we will now outline a proof of Proposition 8.1(i), as given in [14].

8.2. Definition. Let A be some set and let $R \subseteq A^n$ be an n-ary relation. Let $a_1, \ldots, a_{n-1} \in A$ be fixed. Then $\{x \in A \mid R(a_1, \ldots, a_{i-1}, x, a_i, \ldots, a_{n-1})\}$ is called a *section* of R (where $1 \leq i < n$).

8.3. Proposition. (a) *Let $\mathcal{A}' = (\mathbb{N}, 0, S)$. Then*

(i) *$E_{\mathcal{A}'}$ is decidable,*

(ii) *$E_{\mathcal{A}'}$ admits elimination of quantifiers,*

(iii) *a subset $X \subseteq \mathbb{N}$ is definable in $\mathcal{A}'$ iff X is finite or cofinite (i.e., $\mathbb{N} - X$ is finite). More general, every definable n-ary relation $R \subseteq \mathbb{N}^n$ has only finite or cofinite sections.*

(b) *The same as in* (a) *holds for $\mathcal{A} = (\mathbb{N}, 0, S, P)$.*

(c) *Likewise for $(\mathbb{Z}, 0, S, P)$.*

Proof. (a) (see [14]). (i) is proved there by considering the following axiomatization of $E_{\mathscr{A}'}$:

$$S(x) \neq 0,$$

$$S(x) = S(y) \rightarrow x = y,$$

$$y \neq 0 \rightarrow \exists x\, (y = S(x)),$$

$$S(x) \neq x,\ S(S(x)) \neq x, \ldots, S^n(x) \neq x, \ldots \quad \text{(for all } n\text{)}.$$

Using the Loś–Vaught test it is proved that this axiomatization is complete. Obviously it is also decidable. Hence $E_{\mathscr{A}'}$ is decidable.

(ii) As demonstrated in [14], for every assertion $p \in L(\Sigma_{\mathscr{A}'})$ there is a *quantifier-free* assertion q such that $E_{\mathscr{A}'} \vdash p \leftrightarrow q$. (This 'elimination of quantifiers' yields another proof of (i).)

(iii) Routine from (ii).

(b) Note that P is definable in $\mathscr{A}' = (\mathbb{N}, 0, S)$, by

$$P(x) = y \leftrightarrow x = y = 0 \vee S(y) = x.$$

Now use (a).

(c) A routine adaptation of (b). □

8.3.1. Remark. Note that Proposition 8.3(b)(iii) yields an alternative proof of the nondefinability of $+$ in $\mathscr{A}$. For, using a supposed definition of $+$ one could define the set X of even numbers in $\mathscr{A}$; a contradiction since X and its complement are both infinite.

8.4. Application. The following is an example of S_1, S_2 such that the domain inclusion $\mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2)$ is not derivable but can be forced (see Example 9.5(ii)).

Let $\mathscr{A}$ be $(\mathbb{Z}, 0, S, P)$ and $(\Sigma, E) = (\Sigma_{\mathscr{A}}, E_{\mathscr{A}})$. Let

$$S_1 = y := 0;\ \textbf{while}\ x \neq y\ \textbf{do}\ y := S(y)\ \textbf{od};$$
$$y := 0;\ \textbf{while}\ x \neq y\ \textbf{do}\ y := P(y)\ \textbf{od}$$

and

$$S_2 = y := 0;\ \textbf{if}\ x = 0\ \textbf{then}\ x := x\ \textbf{else}\ \mathrm{DIV}\ \textbf{fi}$$

where

$$\mathrm{DIV} = \textbf{while}\ x = x\ \textbf{do}\ x := x\ \textbf{od}.$$

Clearly, S_1 and S_2 converge on $x = 0$ and nowhere else.

Now $\mathrm{HL}(\Sigma, E) \vdash \{x \neq 0\}\, S_2\, \{\textbf{false}\}$, as can easily be proved; however, $\mathrm{HL}(\Sigma, E) \nvdash \{x \neq 0\}\, S_1\, \{\textbf{false}\}$. This can be made plausible by considering an informal proof of $\{x \neq 0\}\, S_1\, \{\textbf{false}\}$; then somehow one must mention the ordering $<$ on $\mathbb{Z}$. However, $<$ is not present in Σ, and not even definable in (Σ, E). (The nondefinability of $<$ in (Σ, E) can easily be proved using Padoa's method (Theorem 3.3), by permuting some of the nonstandard copies of $\mathbb{Z}$ in a nonstandard model of (Σ, E); cf. 3.3.2.)

That $\mathrm{HL}(\Sigma, E) \nvdash \{x \neq 0\}\, S_1\, \{\mathbf{false}\}$ can be made precise as follows. If $\mathrm{HL}(\Sigma, E) \vdash \{x \neq 0\}\, S_1\, \{\mathbf{false}\}$, then, using $x = S(y) \leftrightarrow P(x) = y$, one easily shows that the two invariants $r_1(x, y)$, $r_2(x, y)$ in S_1 must satisfy:

(1) $\quad x \neq 0 \rightarrow r_1(x, 0)$,

(2) $\quad x \neq y \wedge r_1(x, y) \rightarrow r_1(x, S(y))$,

(3) $\quad r_1(x, x) \rightarrow r_2(x, 0)$,

(4) $\quad x \neq y \wedge r_2(x, y) \rightarrow r_2(x, P(y))$,

(5) $\quad \neg r_2(x, x)$.

There are several 'solutions' for r_1, r_2 as subsets of $\mathbb{Z}^2$. However, using (1)–(5) we have $r_1(1, 0)$, hence $r_1(1, 1)$, hence $r_2(1, 0)$, hence $r_2(1, n)$ for all $n \leq 0$. Moreover, from (4) and (5), $\neg r_2(1, m)$ for all $m \geq 1$. Therefore, every solution r_2 has a section which is neither finite nor cofinite; so, by Proposition 8.3(c)(iii), r_2 is not definable.

As promised in Section 7, we will now show that semantical inclusion and forced inclusion are in general not equivalent.

8.5. Theorem. *The proof system* $\mathrm{HL}(\Sigma, E) \Vdash S_1 \sqsubseteq S_2$ *is in general not complete for* $S_1 \sqsubseteq_{\mathrm{Alg}(\Sigma,E)} S_2$.

Proof. Let Σ be the signature of $\mathscr{A} = (\mathbb{N}, 0, S, P)$. From Proposition 8.3(b) we know that $E = E_{\mathscr{A}}$ is decidable. Let $\ulcorner\ \urcorner : \mathscr{WP}(\Sigma) \rightarrow \omega$ be an effective coding of programs; we will write s for $\ulcorner S \urcorner$. R and r are two relations on pairs of codes of programs as follows:

$$r(s_1, s_2) \Leftrightarrow \mathrm{HL}(\Sigma, E) \Vdash S_1 \sqsubseteq S_2,$$

$$R(s_1, s_2) \Leftrightarrow S_1 \sqsubseteq_{\mathrm{Alg}(\Sigma,E)} S_2.$$

The incompleteness of $\Vdash$ for $\sqsubseteq_{\mathrm{Alg}}$ is shown by considering the specification (Σ, E) and demonstrating that $R \neq r$. It turns out that R and r have different positions in the arithmetical hierarchy. As a matter of fact r is Σ_2^0 but R is complete Π_2^0, and a fortiori r and R must differ.

We will first consider r. Working from its formal definition we obtain

$$\begin{aligned} r(S_1, S_2) &\Leftrightarrow \exists (\Sigma', E') \geq (\Sigma, E)\, [\mathrm{HL}(\Sigma, E) \vdash S_1 \sqsubseteq S_2] \\ &\overset{(1)}{\Leftrightarrow} \exists (\Sigma', E') \geq (\Sigma, E)\, [(\Sigma, E) \text{ consistent } \&\ \mathrm{HL}(\Sigma, E) \vdash S_1 \sqsubseteq S_2] \\ &\overset{(2)}{\Leftrightarrow} \exists (\Sigma', E^*)_{\mathrm{finite}}\, [\Sigma' \supseteq \Sigma\ \&\ (\Sigma', E^* \cup E) \text{ consistent} \\ &\qquad \&\ \mathrm{HL}(\Sigma', E^* \cup E) \vdash S_1 \sqsubseteq S_2]. \end{aligned}$$

Step (1) is justified by the completeness of (Σ, E) which entails that each consistent refinement of it is a conservative one. Step (2) follows from Lemma 5.10(ii) which says that the refinement in the definition of $\Vdash$ can be taken finite if one wants. Because '$(\Sigma', E^* \cup E)$ is consistent' is a Π_1^0 predicate and $\mathrm{HL}(\Sigma', E^* \cup E) \vdash S_1 \sqsubseteq S_2$ is Σ_1^0 (due to Theorem 5.11 and the decidability of E), r must be Σ_2^0.

Then consider R. $S_1 \sqsubseteq_{\mathrm{Alg}(\Sigma,E)} S_2$ is in general Π_2^0 in E, R is at most Π_2^0. We have to show that it is complete Π_2^0. A well-known example of a complete Π_2^0 relation is the following one: $t(s) \Leftrightarrow S$ computes a total function on $\mathscr{A}$ (for more information, see [22]). We show that t is 1–1 reducible to R. Let $X_S = \{x_1, \ldots, x_{k(S)}\}$ be the set of variables occurring in S. For $x \in X_S$, $H(x)$ abbreviates the program **while** $x \neq 0$ **do** $x := P(x)$ **od**. $H(X_S)$ abbreviates $H(x_1); H(x_2); \ldots; H(_{k(S)})$. The reduction of t to R works as follows:

$$t(\lceil S \rceil) \Leftrightarrow R(\lceil H(X_S) \rceil, \lceil S; H(X_S) \rceil).$$

To see ($\Leftarrow$), assume $H(X_S) \sqsubseteq_{\mathrm{Alg}(\Sigma,E)} S;\ H(X_S)$; then in $\mathscr{A}$: $H(X_S) \sqsubseteq S;\ H(X_S)$; because $H(X_S)$ is total on $\mathscr{A}$, S must be total on $\mathscr{A}$ as well, i.e., $t(\lceil S \rceil)$ holds. On the other hand assume $t(\lceil S \rceil)$. Let $\mathscr{B} \in \mathrm{Alg}(\Sigma, E)$; clearly $\mathscr{A}$ is isomorphic to a substructure of $\mathscr{B}$. As $H(X_S)$ and $S; H(X_S)$ can only produce output **0** it is sufficient to show $\mathrm{Dom}(H(X_X)) \subseteq \mathrm{Dom}(S; H(X_S))$. $\mathrm{Dom}(H(X_S)) = \mathscr{A}^{k(S)}$, thus S is defined on $\mathrm{Dom}(H(X_S))$ and yields values in $\mathscr{A}^{k(S)}$ on such arguments; on these values in turn, $\mathrm{HL}(X_S)$ is defined. □

9. Domain inclusion

In this section we will show that given some additional information about the domains of S_1, S_2, semantical inclusion and forced inclusion $S_1 \sqsubseteq S_2$ coincide.

9.1. Definition. (i) (*Semantical inclusion of domains*).

Let $S_1, S_2 \in \mathscr{WP}(\Sigma)$. Then $\mathrm{Alg}(\Sigma, E) \models \mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2)$ if, for all $\mathscr{A} \in \mathrm{Alg}(\Sigma, E)$, $\mathrm{Dom}(S_1^{\mathscr{A}}) \subseteq \mathrm{Dom}(S_2^{\mathscr{A}})$. Note that $\mathrm{Alg}(\Sigma, E) \models \mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2)$ implies

$$\mathrm{Alg}(\Sigma, E) \models \{p\}\ S_2\ \{\mathbf{false}\} \Rightarrow \mathrm{Alg}(\Sigma, E) \models \{p\}\ S_1\ \{\mathbf{false}\}.$$

(ii) (HL-*inclusion of domains*). $\mathrm{Dom}(S_1) \sqsubseteq_{\mathrm{HL}(\Sigma,E)} \mathrm{Dom}(S_2)$ iff

$$\mathrm{HL}(\Sigma, E) \vdash \{p\}\ S_2\ \{\mathbf{false}\} \Rightarrow \mathrm{HL}(\Sigma, E) \vdash \{p\}\ S_1\ \{\mathbf{false}\} \quad \text{for all } p \in L(\Sigma).$$

(iii) (*Derivable inclusion of domains*). $\mathrm{HL}(\Sigma, E) \vdash \mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2)$ iff

$$\forall (\Sigma', E') \trianglerighteq (\Sigma, E)\ \ \mathrm{Dom}(S_1) \sqsubseteq_{\mathrm{HL}(\Sigma',E')} \mathrm{Dom}(S_2).$$

(iv) (*Forced inclusion of domains*). $\mathrm{HL}(\Sigma, E) \models \mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2)$ iff

$$\exists (\Sigma', E') \trianglerighteq (\Sigma, E)\ \ \mathrm{HL}(\Sigma', E') \vdash \mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2).$$

9.1.1. Remark. The mathematical theory of domain inclusion is quite complicated in fact. For instance, a pentagon of inclusion relations similar to the one after Theorem 7.1, can be constructed and will turn out to have analogous properties.

In order to prove the main theorem of this section, we need the following proposition.

9.2. Proposition. *Let* $S_1, S_2 \in \mathcal{WP}(\Sigma)$ *contain both the variables* $x_1, \ldots, x_n$ *and suppose* $\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2$. *Then there is a* $(\Sigma', E') \trianglerighteq (\Sigma, E)$ *such that* $\Sigma' \geq \Sigma \cup \{f_1, \ldots, f_n\}$, *where* $f_1, \ldots, f_n$ *are 'fresh' n-ary function symbols, and such that*

$$\mathrm{HL}(\Sigma', E') \vdash \{\boldsymbol{x} = \boldsymbol{z}\}\ S_i\ \{\boldsymbol{x} = f(\boldsymbol{x})\}, \quad i = 1, 2.$$

(Here $\boldsymbol{x} = f(\boldsymbol{z})$ abbreviates: $x_1 = f_1(x_1, \ldots, x_n), \ldots, x_n = f_n(x_1, \ldots, x_n)$.)

Proof. Let $\Sigma'' = \Sigma \cup \{f_1, \ldots, f_n\}$ and $E'' = E \cup \Gamma$ where

$$\Gamma = \{\mathrm{Comp}_{n,S_i}(\boldsymbol{z}) = \boldsymbol{x} \rightarrow \boldsymbol{x} = f(\boldsymbol{z}) \mid n \geq 0, i = 1, 2\}$$

(for 'Comp', see Lemma 1.1.2).

Now every $\mathcal{A} \in \mathrm{Alg}(\Sigma, E)$ can be expanded to an $\mathcal{A}' \in \mathrm{Alg}(\Sigma'', E'')$, since $\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2$. Choose for the interpretation $f^{\mathcal{A}}$ an arbitrary total function extending the partial function $S_2^{\mathcal{A}}$ (which extends itself $S_1^{\mathcal{A}}$). Therefore, by the criterion for conservativity (Proposition 2.7.1), $(\Sigma'', E'') \trianglerighteq (\Sigma, E)$. Clearly, $\mathrm{Alg}(\Sigma'', E'') \models \{\boldsymbol{x} = \boldsymbol{z}\}\ S_i\ \{\boldsymbol{x} = f(\boldsymbol{z})\}$, $i = 1, 2$.

Now let (Σ', E') be a logical completion of (Σ'', E''). (By Theorem 6.1 this exists.) Then $\mathrm{Alg}(\Sigma', E') \models \{\boldsymbol{x} = \boldsymbol{z}\}\ S_i\ \{\boldsymbol{x} = f(\boldsymbol{z})\}$, $i = 1, 2$, and by the logical completeness we have

$$\mathrm{HL}(\Sigma', E') \vdash \{\boldsymbol{x} = \boldsymbol{z}\}\ S_i\ \{\boldsymbol{x} = f(\boldsymbol{z})\}. \quad \square$$

9.3. Theorem. *Suppose* $HL(\Sigma, E) \Vdash \mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2)$. *Then*

$$\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2 \Leftrightarrow \mathrm{HL}(\Sigma, E) \Vdash S_1 \sqsubseteq S_2.$$

Proof. ($\Leftarrow$) is already done in Section 7.

($\Rightarrow$). Let $S_1, S_2 \in \mathcal{WP}(\Sigma)$ be such that

$$\mathrm{HL}(\Sigma, E) \Vdash \mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2) \quad \text{and} \quad \mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2.$$

Let $\boldsymbol{x} = x_1, \ldots, x_n$ be the variables occurring in S_1, S_2.

Step 1. Extend Σ to Σ_1 containing n-ary function symbols $f_1, \ldots, f_n$ and E to E_1 such that $(\Sigma_1, E_1) \trianglerighteq (\Sigma, E)$ and $\mathrm{HL}(\Sigma_1, E_1) \vdash \{\boldsymbol{x} = \boldsymbol{z}\}\ S_i\ \{\boldsymbol{x} = f(\boldsymbol{z})\}$, $i = 1, 2$. This is possible by Proposition 8.2.

By assumption, there is a $(\Sigma_2, E_2) \trianglerighteq (\Sigma, E)$ such that $\mathrm{HL}(\Sigma_2, E_2) \vdash \mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2)$. We may suppose $\Sigma_2 \cap \Sigma_1 = \Sigma$ (cf. Proposition 4.7.2), hence by Robinson's

Consistency Theorem 2.6.2, $(\Sigma', E') = (\Sigma_1 \cup \Sigma_2, E_1 \cup E_2)$ is a conservative refinement of (Σ, E).

Claim. $\mathrm{HL}(\Sigma', E') \vdash S_1 \sqsubseteq S_2$. (Then we are through.)

Proof of the Claim. Consider a refinement $(\Sigma'', E'') \trianglerighteq (\Sigma', E')$ such that

$$\mathrm{HL}(\Sigma'', E'') \vdash \{p\}\, S_2\, \{q\}.$$

We have to prove

$$(0) \qquad \mathrm{HL}(\Sigma'', E'') \vdash \{p\}\, S_1\, \{q\}.$$

Obviously, since $q[f(\boldsymbol{x})/(\boldsymbol{x}] \wedge \neg q\,[f(\boldsymbol{x})/\boldsymbol{x}]$ is a tautology, (0) is equivalent with (1) & (2) as follows:

$$(1) \qquad \mathrm{HL}(\Sigma'', E'') \vdash \{p \wedge q[f(\boldsymbol{x})/\boldsymbol{x}]\}\, S_1\, \{q\},$$

$$(2) \qquad \mathrm{HL}(\Sigma'', E'') \vdash \{p \wedge \neg q[f(\boldsymbol{x})/\boldsymbol{x}]\}\, S_1\, \{q\}.$$

Proof of (1). By the rule of consequence, it is sufficient to prove that

$$\mathrm{HL}(\Sigma'', E'') \vdash \{q[f(\boldsymbol{x})/\boldsymbol{x}]\}\, S_1\, \{q\}.$$

We know that

$$\mathrm{HL}(\Sigma_1, E_1) \vdash \{\boldsymbol{x} = \boldsymbol{z}\}\, S_1\, \{\boldsymbol{x} = f(\boldsymbol{z})\},$$

hence, trivially,

$$\mathrm{HL}(\Sigma'', E'') \vdash \{\boldsymbol{x} = \boldsymbol{z}\}\, S_1\, \{\boldsymbol{x} = f(\boldsymbol{z})\}.$$

By Proposition 1.2.3 it follows that

$$\mathrm{HL}(\Sigma'', E'') \vdash \{\boldsymbol{x} = \boldsymbol{z} \wedge q[f(\boldsymbol{z})/\boldsymbol{z}]\}\, S_1\, \{\boldsymbol{x} = f(\boldsymbol{z}) \wedge q[f(\boldsymbol{z})/\boldsymbol{z}]\}.$$

Hence indeed $\mathrm{HL}(\Sigma'', E'') \vdash \{q[f(\boldsymbol{x})/\boldsymbol{x}]\}\, S_1\, \{q\}$.

Proof of (2). We know that $\mathrm{HL}(\Sigma'', E'') \vdash \{p\}\, S_2\, \{q\}$. So, by the Conjunction rule (1.2.3(i)) and Invariance rule (1.2.3(iii)) we have

$$\mathrm{HL}(\Sigma'', E'') \vdash \{\boldsymbol{x} = \boldsymbol{z} \wedge p \wedge \neg q[f(\boldsymbol{z})/\boldsymbol{x}]\}\, S_2\, \{q \wedge x = f(\boldsymbol{z}) \wedge \neg q[f(\boldsymbol{z})/\boldsymbol{x}]\}$$

where the postcondition obviously implies $\{\mathbf{false}\}$. By the assumption $\mathrm{HL}(\Sigma_2, E_2) \vdash \mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2)$ we have, therefore, the same for S_1:

$$\mathrm{HL}(\Sigma'', E'') \vdash \{\boldsymbol{x} = \boldsymbol{z} \wedge p \wedge \neg q[f(\boldsymbol{z})/\boldsymbol{x}]\}\, S_1\, \{\mathbf{false}\}.$$

By the rule of consequence we have

$$\mathrm{HL}(\Sigma'', E'') \vdash \{\boldsymbol{x} = \boldsymbol{z} \wedge p \wedge \neg q[f(\boldsymbol{z})/\boldsymbol{x}]\}\, S_1\, \{q\}.$$

By Proposition 1.2.3(iv) we have

$$\mathrm{HL}(\Sigma'', E'') \vdash \{\exists \boldsymbol{z}(\boldsymbol{x} = \boldsymbol{z} \wedge p \wedge \neg q[f(\boldsymbol{z})/\boldsymbol{x}]\}\, S_1\, \{q\}.$$

I.e., indeed $\mathrm{HL}(\Sigma'', E'') \vdash \{p \wedge \neg q[f(\boldsymbol{x})/\boldsymbol{x}]\}\, S_1\, \{q\}$. $\square$

9.4. Corollary. *Let $S_1, S_2 \in \mathcal{WP}(\Sigma)$ and suppose that S_2 is everywhere converging, for all $\mathcal{A} \in \mathrm{Alg}(\Sigma, E)$. Then*

$$\mathrm{Alg}(\Sigma, E) \models S_1 \sqsubseteq S_2 \Leftrightarrow \mathrm{HL}(\Sigma, E) \Vdash S_1 \sqsubseteq S_2.$$

Proof. ($\Leftarrow$) has already been proved in Section 7.

($\Rightarrow$). By the soundness of HL (Lemma 1.2.1) we see that $\mathrm{HL}(\Sigma, E) \nvdash \{p\}\ S_2\ \{\mathbf{false}\}$ for all $p \in L(\Sigma)$. Hence trivially

$$\mathrm{HL}(\Sigma, E) \vdash \{p\}\ S_2\ \{\mathbf{false}\} \Rightarrow \mathrm{HL}(\Sigma, E) \vdash \{p\}\ S_1\ \{\mathbf{false}\},$$

i.e., $\mathrm{HL}(\Sigma, E) \vdash \mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2)$.

Therefore, also trivially, $\mathrm{HL}(\Sigma, E) \Vdash \mathrm{Dom}(S_1) \sqsubseteq \mathrm{Dom}(S_2)$. Now apply the preceding theorem. □

9.5. Example. (i) Let S_1, S_2 be as in Example 7.5. Then $\mathrm{HL}(\Sigma_{\mathcal{A}}, E_{\mathcal{A}}) \Vdash S_1 \sqsubseteq S_2$ and S_2 is always converging. Hence by 8.4, $\mathrm{Alg}(\Sigma_{\mathcal{A}}, E_{\mathcal{A}}) \models S_1 \sqsubseteq S_2$.

(ii) In Example 9.5(i) the domain inclusion is already derivable. An example where domain inclusion is not derivable but can be forced, was given in 8.4.

References

[1] K.R. Apt, Ten years of Hoare's logic—A survey, in: F.V. Jensen, B.H. Mayoh and K.K. Møller, eds., *Proc. 5th Scandinavian Logic Symp.* (Aalborg University Press, Aalborg, 1979) pp. 1–44.

[2] R.J. Back, *Correctness Preserving Program Refinements: Proof Theory and Applications*, Mathematical Centre Tracts **131** (Mathematical Centre, Amsterdam, 1980).

[3] J.W. De Bakker, *Recursive Procedures*, Mathematical Centre Tracts **24** (Mathematical Centre, Amsterdam, 1973).

[4] J.W. De Bakker, *Mathematical Theory of Program Correctness* (Prentice-Hall, London, 1980).

[5] J.A. Bergstra, J. Tiuryn and J.V. Tucker, Floyd's principle, correctness theories and program equivalence, *Theoret. Comput. Sci.* **17** (1982) 113–149.

[6] J.A. Bergstra and J. Terlouw, A characterisation of program equivalence in terms of Hoare's logic, in: *Proc. G.I. Jahrestagung*, München, 1981, Lecture Notes in Comput. Sci. **123** (Springer, Berlin, 1981).

[7] J.A. Bergstra and J.V. Tucker, Expressiveness and the completeness of Hoare's logic, J. Comput. System Sci. **25** (1982) 267–284.

[8] J. Bergstra and J.V. Tucker, The refinement of specifications and the stability of Hoare's logic, in: D. Kozen, ed., *Logics of Programs*, Lecture Notes in Comput. Sci. **131** (Springer, Berlin, 1982) pp. 24–36.

[9] J.A. Bergstra and J.V. Tucker, Hoare's logic and Peano's arithmetic, *Theoret. Comput. Sci.* **22** (1983) 265–284.

[10] J.A. Bergstra and J.V. Tucker, Two theorems about the completeness of Hoare's logic, *Inform. Process. Lett.* **15** (1982) 143–149.

[11] G.S. Boolos and R.C. Jeffrey, *Computability and Logic* (Cambridge University Press, 1974/1980).

[12] E.M. Clarke, Programming language constructs for which it is impossible to obtain good Hoare-like axioms, *J. Assoc. Comput. Mach.* **26** (1979) 129–147.

[13] S.A. Cook, Soundness and completeness of an axiom system for program verification, *SIAM J. Comput.* **7** (1978) 70–90.

[14] H.B. Enderton, *A Mathematical Introduction to Logic* (Academic Press, New York, 1972).

[15] D. Harel, A. Pnueli and J. Stavi, A complete axiom system for proving deduction about recursive programs, in: *Proc. 9th ACM Symp. on Theory of Computing*, Boulder, 1977.

[16] C. Hemerik, Relaties tussen taaldefinitie en taalimplementatie, in: J.C. van Vliet, red., *Coll. Capita Implementatie van Programmeertalen*, MC. Syllabus **42** (Mathematical Centre, Amsterdam, 1980) (in Dutch).

[17] C.A.R. Hoare and P. Lauer, Consistent and complementary formal theories of the semantics of programming languages, *Acta Inform.* **3** (1974) 135–155.

[18] C.A.R. Hoare, An axiomatic basis for computer programming, *Comm. ACM* **12** (1967) 576–580.

[19] J. Lambek, How to program an infinite abacus, *Canad. Math. Bulletin* **4** (1961) 295–302.

[20] Z. Manna, *Mathematical Theory of Computation* (McGraw-Hill, New York, 1974).

[21] J.D. Monk, *Mathematical Logic* (Springer, Berlin, 1976).

[22] H. Rogers, *Theory of Recursive Functions and Effective Computability* (McGraw-Hill, New York, 1967).

[23] B. Russell, Correctness of the compiling process based on axiomatic semantics, *Acta Inform.* **14** (1980) 1–20.

[24] J. Shoenfield, *Mathematical Logic* (Addison-Wesley, Reading, MA, 1967).

[25] M. Wand, A new incompleteness result for Hoare's system, *J. Assoc. Comput. Mach.* **25** (1978) 168–175.