stichting
mathematisch
centrum

MC

AFDELING INFORMATICA
(DEPARTMENT OF COMPUTER SCIENCE)

IW 165/81 APRIL

J.A. BERGSTRA & J.V. TUCKER

TWO THEOREMS ABOUT THE COMPLETENESS OF HOARE'S LOGIC

Preprint

kruislaan 413 1098 SJ amsterdam

Printed at the Mathematical Centre, 413 Kruislaan, Amsterdam.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications. It is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O.).

1980 Mathematics subject classification: 03D45, 03D80, 68B15, 03D35, 03D75, 68B10

ACM - Computing Review-category 4.34, 5.24

Two theorems about the completeness of Hoare's logic*

by

J.A. Bergstra** & J.V. Tucker***

ABSTRACT

We prove two theorems about the completeness of Hoare's logic for the partial correctness of <u>while</u> programs over an axiomatic specification. The first result is a completion theorem: any specification (Σ,E) can be refined to a specification (Σ_0,E_0), conservative over (Σ,E), whose Hoare's logic is complete. The second result is a normal form theorem: any complete specification (Σ,E) possessing some complete logic for partial correctness can be refined to an effective specification (Σ_0,E_0), conservative over (Σ,E), which generates all true partial correctness formulae with Hoare's standard rules.

KEY WORDS & PHRASES: *Hoare's logic, partial correctness, while-programs, data type specifications, refinements, strongest postcondition calculus, Peano arithmetic, logical completeness*

---

* This paper is not for review as it will be published elsewhere.

** Department of Computer Science, University of Leiden, Wassenaarseweg 80, Postbus 9512, 2300 RA LEIDEN, The Netherlands

*** School of Mathematics and Computer Science, University of Bristol, University Walk, Bristol BS8 1TW, England

INTRODUCTION

With the term Hoare's logic we mean the formal system for the manupulation of statements about the partial correctness of while-programs first described in HOARE [13] and studied in COOK [10]. It is a two tiered axiomatic system with axioms and proof rules for asserted programs linked bu the Rule of Consequence to a conventional axiomatic theory which generates first-order assertions about the class of data structures on which the programs compute. In this note we will prove a theorem about the completeness of the Hoare's logic built from *any* axiomatic specification, and another theorem which suggests that Hoare's rules provide a system which is generic among all possible logics for partial correctness.

Let (Σ,E) be some axiomatic specification where Σ is a finite signature and E a set of axioms written in $L(\Sigma)$, the first-order language over Σ. Let $HL(\Sigma,E)$ be the set of $L(\Sigma)$-asserted programs provable in Hoare's logic for (Σ,E). Let $PC(\Sigma,E)$ be the set of all $L(\Sigma)$-asserted programs true in *all* models of (Σ,E). The soundness of a Hoare's logic is simply the inclusion $HL(\Sigma,E) \subset PC(\Sigma,E)$. Let us say that the Hoare's logic is *logically complete* if $HL(\Sigma,E) = PC(\Sigma,E)$. This notion of completeness is the natural proof-theoretical choice (one thinks of the Completeness Theorem for first-order logic)and is a companion to Cook's semantical notion of completeness in [10] which is based upon validity in a particular model of the specifying theory. There is no entirely *general* completeness theorem for Hoare's logic. To take arithmetic, for example, *only* the Hoare logic made from complete number number theory is logically complete [7]. However, we prove the following "completion theorem" in Section 3. (We assume throughout this paper that (Σ,E) is a specification having no finite models.)

THEOREM. *Any axiomatic specification* (Σ,E) *can be refined to a specification* (Σ_0,E_0) *which proves precisely the same* $L(\Sigma)$ *assertions and yet possesses a logically complete Hoare logic* $HL(\Sigma_0,E_0)$.

A specifying theory (Σ,E) is complete if any $L(\Sigma)$ assertion can be decided from the axioms; given any sentence $p \in L(\Sigma)$either $E \vdash p$ or $E \vdash \neg p$. Our other result is this "normal form theorem" of Section 4.

<u>THEOREM</u> *Any complete specification* (Σ,E) *possessing some complete logic for partial correctness can be refined to an effective specification* (Σ_0,E_0) *which proves precisely the same* $L(\Sigma)$ *assertions and yet* $PC(\Sigma,E) \subset HL(\Sigma_0,E_0)$.

This note is a companion to our paper [7] which is part of a series about Hoare's logic and its proof theory [5,6,7,9]; see also [3,4]. Obviously we are assuming readers to be familiar with the papers HOARE [13] and COOK [10]; some other material we require will be carefully documented in preliminary Sections 1 and 2. The invaluable survey APT[1] is also recommended.

1. ASSERTIONS, SPECIFICATIONS AND PROGRAMS.

<u>SYNTAX</u>. Let Σ be a finite signature - all signatures in this paper are finite. Let $L(\Sigma)$ be the first-order logical language with equality based on Σ. Let E be a set of (the universal closures of) assertions of $L(\Sigma)$; the pair (Σ,E) is a theory or, as we prefer in the present context, a *specification*. The set of all theorems in $L(\Sigma)$ provable from E is denoted $Thm(\Sigma,E)$; we often write $E \vdash p$ for $p \in Thm(\Sigma,E)$ when $p \in L(\Sigma)$ is understood.

A specification (Σ,E) is *complete* if for any sentence $p \in L(\Sigma)$ either $E \vdash p$ or $E \vdash \neg p$.

A specification (Σ',E') is a refinement of specification (Σ,E) if $\Sigma \subset \Sigma'$ and $Thm(\Sigma,E) \subset Thm(\Sigma',E')$. Two specifications are *logically equivalent* if each refines the other.

The specification (Σ',E') is a *conservative refinement* of (Σ,E) if (Σ',E') is a refinement of (Σ,E) in which for any $p \in L(\Sigma)$

$$E' \vdash p \quad \text{implies} \quad E \vdash p$$

The set of all <u>while</u>-programs based on Σ is defined in the usual way using the syntax of $L(\Sigma)$ and is denoted $\mathcal{WP}(\Sigma)$. By a *specified* or *asserted program* we mean a triple of the form $\{p\}S\{q\}$, where $S \in \mathcal{WP}(\Sigma)$ and $p,q, \in L(\Sigma)$.

<u>SEMANTICS</u>. The semantics of $L(\Sigma)$ is the satisfaction semantics of model theory. The validity of assertion $p \in L(\Sigma)$ for structure A we write $A \models p$. The class of all models of a specification (Σ,E) is denoted $Mod(\Sigma,E)$ or

simply Mod(E)when Σ is clearly understood in the context. For $p \in L(\Sigma)$, we write Mod(E) $\models$ p to mean for every $A \in$ Mod(E), $A \models p$. As far as any proof therory of a data type specification is concerned, the semantics of a specification (Σ,E) is Mod(Σ,E):

1.1 COMPLETENESS THEOREM. *Let* (Σ,E) *be a specification. For* $P \in L(\Sigma)$ $E \vdash p$ *if, and only if,* Mod(E) $\vdash$ p.

For the semantics of $\mathcal{WP}(\Sigma)$as determined by a structure A we leave the reader free to choose any sensible account of while-program computations which applies to an arbitrary structure. COOK [10]; the graph-theoretic semantics in GREIBACH [12]; the denotational semantics described in DE BAKKER [2].

To the asserted programs we assign *partial correctness semantics*: the asserted program {p}S{q}is valid on a structure A if for each initial state $a \in$ States(A), $A \models p(a)$ implies either S(a)terminates and $A \models q(S(a))$or S(a) diverges; in symbols, $A \models$ {p}S{q}. And the asserted program {p}S{q}is valid for a specification E if it is valid on every model of E; in symbols, Mod(E) $\models$ {p}S{q}.

The *partial correctness theory of a structure* A is the set

$$PC(A) = \{\{p\}S\{q\}: A \models \{p\}S\{q\}\}$$

and the *partial correctness theory of a specification*(Σ,E) is the set

$$PC(\Sigma,E) = \{\{p\}S\{q\};\ Mod(\Sigma,E) \models \{p\}S\{q\}\}.$$

The strongest postcondition of $S \in \mathcal{WP}(\Sigma)$and $p \in L(\Sigma)$on structure A is the set

$$SP_A(p,S) = \{b \in \text{States } A;\ \exists a \in \text{States}(A)[S(a) \text{ terminates in final state } b \text{ and } A \models p(a)]\}$$

1.2. LEMMA. $A \models \{p\}S\{q\}$ *if, and only if,* $SP_A(p,S) \subset \{b \in \text{States}(A): A \models q(b)\}$.

Let A be a structure of signature Σ. We say $L(\Sigma)$is *expressive* for $\mathcal{WP}(\Sigma)$over A if for each $S \in \mathcal{WP}(\Sigma)$and $p \in L(\Sigma)$the strongest postcondition $SP_A(p,S)$ is definable by an assertion of $L(\Sigma)$.

PEANO ARITHMETIC AND INDUCTIVE REFINEMENTS. Let N be the standard model of arithmetic with primitive operations the successor function x+1, addition x+y, multiplication x.y; and with 0 as distinguished constant. We shall use these notations for the functions and the functions symbols of its signature Σ_N.

Peano arithmetic PA is built up as follows:

Operator axioms: 1. $0 \neq x+1$

2. $x+1 = y+1 \rightarrow x = y$
3. $x+0 = x$
4. $x+(y+1) = (x+y)+1$
5. $x.0 = 0$
6. $x.(y+1) = x.y + x$

Induction scheme: for each assertion $p \in L(\Sigma_N)$, containing free variable x, the following is an axiom

$$p(0) \wedge \forall x[p(x) \rightarrow p(x+1)] \rightarrow \forall x.p(x)$$

That this simple axiomatic description of arithmetic captures all but the more esoteric properties of N makes it a natural object of study in the logic of programs[9]. But adding PA to a specification turns out to be a rather important idea, too (as confirmed by Basic Lemma 2.5).

A specification (Σ,E) is an *inductive refinement of Peano arithmetic* if it is a refinement of PA and it allows induction in the following form: for any $\phi(x) \in L(\Sigma)$ with free variable x

$$E \vdash \phi(0) \wedge \forall x[\phi(x) \rightarrow \phi(x+1)] \rightarrow \forall x.\phi(x)$$

A model A of (Σ,E) is called *standard* if the Σ_N-reduct of A, $A|_{\Sigma_N}$, is isomorphic to N.

For *any* specification (Σ,E) it is obviously of some interest to look at the minimal inductive refinement of Peano and (Σ,E) made by adjoining Σ_N to E and PA to E and closing with the induction scheme over $\Sigma \cup \Sigma_N$, if necessary. Let this specification be called the *Peano companion* of (Σ,E) and denote it $PA(\Sigma,E)$.

1.3. LEMMA *Let* (Σ,E) *be a specification. If* A *is a* $\Sigma\cup\Sigma_N$*-structure whose* Σ*-reduct satisfies the axioms* E *and whose* Σ_N*-reduct is isomorphic to* N *then* A *is a standard model of* PA(Σ,E)

2. HOARE'S LOGIC

Hoare's logic for while-programs over specification (Σ,E) with first-order assertion language $L(\Sigma)$has the usual axioms and proof rules and there can be found in HOARE [13], COOK [10], DE BAKKER [2], or APT [1]. But needing an explicit citation is the rule of inference called the *Consequence Rule*: for $S \in \mathcal{WP}(\Sigma), p, q, p_1, q_1 \in L(\Sigma)$

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

and, in connection with it, the *oracle* of axioms: *Each member of* Thm(Σ,E) *is an axiom.* The set of all asserted programs provable in Hoare's logic for (Σ,E) we denote HL(Σ,E) and we write HL$(\Sigma,E) \vdash \{p\}S\{q\}$in place of $\{p\}S\{q\} \in$ HL(Σ,E). The following fact is obvious.

2.1. REFINEMENT LEMMA.*Let* (Σ,E) *and* (Σ',E') *be specifications. If* (Σ',E') *is a refinement of* (Σ,E) *then* HL$(\Sigma,E) \subset$ HL(Σ',E'). *Thus, if* (Σ,E)*and* (Σ',E') *are equivalent specifications then* HL(Σ,E) = HL(Σ',E').

The Corollary to Theorem 1 in COOK [10] can be stated as follows.

2.2 SOUNDNESS THEOREM. *For any specification* (Σ,E), HL$(\Sigma,E) \subset$ PC(Σ,E).

The Hoare's logic for specification $(\Sigma,E)$is said to be *logically complete* if HL(Σ,E) = PC(Σ,E).

The completeness result devised in COOK [10] can be stated as follows.

2.3 COOK'S COMPLETENESS THEOREM; *Let* (Σ,E)*be a complete specification with model* A. *If* $L(\Sigma)$*is expressive for* $\mathcal{WP}(\Sigma)$*over* A *then* HL(Σ,E) = PC(A).

In contrast to our notion of logical completeness which is a specification invariant and which derives from the Completeness Theorem 1.1, the notion of adequacy involved in Theorem 2.3 depends upon specification and a particular model. Actually, the strength of the completeness assumption on the specification is enough to fuse the independent approaches [7]:

2.4 <u>THEOREM</u> *Let* (Σ,E) *be a complete specification. If* (Σ,E) *possesses a model* A *for which* $L(\Sigma)$ *is expressive for* $\mathcal{WP}(\Sigma)$ *then* $HL(\Sigma,E) = PC(\Sigma,E)$ *- the Hoare's logic of* (Σ,E) *is logically complete.*

Complete specifications do not always provide logically complete Hoare logics; Presburger arithmetic illustrates this [10,5]. On the other hand incomplete specifications can provide logically complete Hoare logics; this is a valuable corollary of our Theorem 3.1 here.

Although expressiveness is not a proof theoretical notion (it is not preserved by elementary equivalence [7]) its rôle in structural completeness is echoed in the present concern with logical completeness. The following theorem about Peano refinements is extracted from our [9].

2.5 <u>BASIC LEMMA</u>. *Let* (Σ,E) *be an inductive refinement of Peano arithmetic. Given any assertion* $p \in L(\Sigma)$ *and program* $S \in \mathcal{WP}(\Sigma)$ *one can effectively calculate an assertion* $SP(p,S) \in L(\Sigma)$ *such that*

1. $HL(\Sigma,E) \vdash \{p\}S\{SP(p,S)\}$
2. $HL(\Sigma,E) \vdash \{p\}S\{q\}$ *if, and only if,* $E \vdash SP(p,S) \rightarrow q$
3. *Over each standard model* A *of* (Σ,E) *the formula* $SP(p,S)$ *defines the strongest postcondition* $SP_A(p,S)$.

It should be noted that Basic Lemma 2.5 provides an entirely proof theoretical representation of the strongest postcondition calculus: statements (1) and (2) are responsible for the significance of the formula; statement (3) is a semantic accessory so to say.

We conclude our preliminaries with some remarks on logics for partial correctness.

Quite obviously, for any specification (Σ,E), $HL(\Sigma,E)$ is recursively enumerable in $Thm(\Sigma,E)$. Taking the weakest criterion one can sensibly use, we define any set $\ell pc(\Sigma,E)$ of asserted programs which is r.e. in $Thm(\Sigma,E)$

to be a *logic of partial correctness* for the specification (Σ,E).

A logic of partial correctness $\ell pc(\Sigma,E)$ is *sound* if $\ell pc(\Sigma,E) \subset PC(\Sigma,E)$ and is *logically complete* if $\ell pc(\Sigma,E) = PC(\Sigma,E)$.

Following [5], it is easy to prove that $PC(\Sigma,E)$ is co-r.e in $Thm(\Sigma,E)$; we have the following theorem in consequence.

2.6. LEMMA. *There exists a sound and logically complete logic of partial correctness* $\ell pc(\Sigma,E)$ *for a specification* (Σ,E) *if, and only if,* $PC(\Sigma,E)$ *is recursive in* $Thm(\Sigma,E)$.

3. A COMPLETION THEOREM

In this section we prove the following completion theorem:

3.1 THEOREM *Let* (Σ,E) *be a specification. Then there is a conservative refinement* (Σ_0,E_0) *of* (Σ,E) *for which* $HL(\Sigma_0,E_0)$ *is logically complete.*

PROOF. Let $\Sigma_0 = \Sigma \cup \Sigma_N$. For each countable ordinal α we inductively define a set T_α of assertions from $L(\Sigma_0)$ using Basic Lemma 2.5: for the basis,

$$T_0 = PA(\Sigma,E)$$

for each countable ordinal α,

$$T_{\alpha+1} = T_\alpha \cup \{SP(p,S) \rightarrow q\colon p,q \in L(\Sigma_0), S \in \mathcal{WP}(\Sigma) \text{ and } Mod(T_\alpha) \models \{p\}S\{q\}\}$$

for each countable limit ordinal γ,

$$T_\lambda = \bigcup_{\beta<\gamma} T_\beta \ .$$

Clearly, the countability of $L(\Sigma_0)$ entails that for some countable ordinal α, $T_\gamma = T_{\gamma+1}$. Let σ be the least such ordinal, the *degree* of (Σ,E), and set $E_0 = T_\sigma$.

3.2 LEMMA. $HL(\Sigma_0,E_0)$ *is logically complete.*

PROOF. Suppose $\mathrm{Mod}(E_0) \models \{p\}S\{q\}$. Let γ be the least ordinal index such that $\mathrm{Mod}(T_\gamma) \models \{p\}S\{q\}$. Then, obviously, $\gamma \leq \sigma$ and, by construction, we know that

$$SP(p,S) \to q \in T_{\gamma+1} \subset E_0.$$

By Basic Lemma 2.5,

$$HL(\Sigma_0,E_0) \vdash \{p\}S\{SP(p,S)\}$$

Thus, by the Rule of Consequence,

$$HL(\Sigma_0,E_0) \vdash \{p\}S\{q\}$$

and we are done. □

3.3 LEMMA. (Σ_0,E_0) *is conservative over* (Σ,E)

PROOF. Suppose for a contradiction that p is an assertion of $L(\Sigma)$ such that

$$E_0 \vdash p \quad \text{but} \quad E \nvdash p .$$

By the Completeness Theorem 1.1, there must exist a model A of signature Σ for $E \cup \{\neg p\}$. By the Downward Löwenheim-Skolem Theorem, we may assume A to be countable and hence we may choose functions on A to interpret the new symbols of Σ_N such that the augmented structure B is a standard model of $PA(\Sigma_0,E_0)$; in symbols, $B|_{\Sigma_N} \cong N$.

We shall prove that $B \models E_0$. Once this is done we may observe that $B \models E_0 \cup \{\neg p\}$ and so $E_0 \nvdash p$, the required contradiction.

We show that $B \models E_0$ by induction on the ordinals indexing the construction of E_0. The basis follows from Lemma 1.3. Assume $B \models T_\alpha$ and consider $T_{\alpha+1}$. Let $SP(p,S) \to q \in T_{\alpha+1} - T_\alpha$. Then $\mathrm{Mod}(T_\alpha) \models \{p\}S\{q\}$ and so $B \models \{p\}S\{q\}$. By Basic Lemma 2.5, we know $B \models SP(p,S) \to q$. Therefore, $B \models T_{\alpha+1}$. Lastly, if $B \models T_\beta$ for each $\beta < \gamma$ then $B \models \bigcup_{\beta<\gamma} T_\beta$ and this is $B \models T_\gamma$. □

3.4. COROLLARY. *Incomplete specifications may have logically complete Hoare logics.*

An obvious, but important, question arising from this argument is the following.

3.5 QUESTION *Can the set of axioms* E_0 *be proved, or chosen, to be a not too complicated set, for example an arithmetical set, or even a recursively enumerable set in certain circumstances?*

The degree σ of the specification (Σ,E) is an unknown quantity.

4. A NORMAL FORM THEOREM.

In this section we prove the following normal form theorem for Hoare-like logics.

4.1. THEOREM. *Let* (Σ,E) *be a specification which is complete. If the partial correctness theory* $PC(\Sigma,E)$ *possesses a complete logic* $\ell pc(\Sigma,E)$ *then there is a recursive and conservative refinement* (Σ_0,E_0) *of* (Σ,E) *for which the standard Hoare logic* $HL(\Sigma_0,E_0)$ *contains* $PC(\Sigma,E)$.

PROOF. From the definition of a logic of partial correctness and Lemma 2.5, we know that $PC(\Sigma,E)$ is recursive in $Thm(\Sigma,E)$. Formally, let $\{\{p_i\}S_i\{q_i\}: i \in \omega\}$ be an enumeration of all asserted programs with $p_i,q_i \in L(\Sigma)$ and $S_i \in \mathcal{WP}$. Let $\{\phi_i: i \in \omega\}$ be an enumeration of all assertions of $L(\Sigma)$provable from E. The assumption that $PC(\Sigma,E)$ is recursively enumerable in Thm (Σ,E) means that $A = \{i \in \omega: \{p_i\}S_i\{q_i\} \in PC(\Sigma,E)\}$ is r.e. in $B = \{i \in \omega: \phi_i \in Thm(\Sigma,E)\}$ and we can claim

4.2 LEMMA. *There is a recursive function* $f: \omega \to \omega$ *such that* $f(B) = A$.

PROOF. Let for $n \in \omega$ D_n be the uniquely determined set $\{a_1,\ldots,a_k\}$ such that $n = 2^{a_1} + \ldots + 2^{a_k}$ and $a_1 < \ldots < a_k$, if $n > 0$ and $D_n = \emptyset$ if $n = 0$. Denoting with T the formula $\forall x(x=x)$ a formula $F(n,m,\ell,i)$ is defined as follows:

$$F(n,m,\ell,i) \equiv \bigwedge_{j\in D_n} \phi_j \wedge \bigwedge_{j\in D_m} \neg\phi_j \wedge \Big(\bigwedge_{\ell\text{-times}} T \vee \bigwedge_{i\text{-times}} T\Big)$$

It is meant that from $F(n,m,\ell,i)$ one can read off n,m,ℓ and i immediately. Moreover, if $D_n \subseteq B$ and $D_m \subseteq \bar{B}$ $F(n,m,b,i) \in B$

Because A is r.e. in B there exists a recursive function g such that

$$i \in A \Longleftrightarrow \exists n,m,\ell(D_n \subseteq B \wedge D_m \subseteq \bar{B} \wedge g(n,m,\ell,i) = 0).$$

The function f can now be given, after choosing k_0 to be some fixed element of A.

$f(j)$ = if ϕ_j is of the form $F(n,m,\ell,i)$ for some $n,m,\ell,i \in \omega$, (necessarily uniquely determined) and $g(n,m,\ell,i) = 0$ then i, otherwise k_0.

It is not hard to verify that this f works.

With the construction of Theorem 3.1 in mind, and the association of $\{p_{f(i)}\}S_{f(i)}\{q_{f(i)}\}$ to ϕ_i, we set $\Sigma_0 = \Sigma \cup \Sigma_N$ and define

$$E_0 = PA(\Sigma,E) \cup \{SP(\phi_i \wedge p_{f(i)},S_{f(i)}) \rightarrow q_{f(i)} : i \in \omega\}$$

Obviously, (Σ_0,E_0) is an r.e. refinement of (Σ,E); we have to show that its Hoare's logic embraces $PC(\Sigma,E)$ and that it is conservative.

4.3 <u>LEMMA</u>. $PC(\Sigma,E) \subset HL(\Sigma_0,E_0)$

<u>PROOF</u>. Let $\{p_j\}S_j\{q_j\} \in PC(\Sigma,E)$ and choose some $i \in B$ such that $f(i) = j$. Because $\phi_i \in Thm(E)$ we have

$$E_0 \vdash p_j \rightarrow \phi_i \wedge p_j$$

and by definition

$$E_0 \vdash SP(\phi_i \wedge p_j,S_j) \rightarrow q_j$$

By the Basic Lemma 2.5 about inductive refinements,

$$HL(\Sigma_0,E_0) \vdash \{p_j\}S_j\{q_j\}. \qquad \square$$

To show that (Σ_0,E_0) is a conservative refinement of (Σ,E) we need only show that E_0 is refined by the theory T_1, the second stage in the construction of T_σ in the proof of Theorem 3.1. Remember that

$$T_1 = PA(\Sigma,E) \cup \{SP(p,S) \rightarrow q\colon Mod(PA(\Sigma,E)) \models \{p\}S\{q\}\}$$

Clearly, it is sufficient to check that for $i \in \omega$

$$Mod(PA(\Sigma,E)) \models \{\phi_i \wedge p_{f(i)}\}\, S_{f(i)}\{q_{f(i)}\}$$

in which case $E_0 \subset T_1$.

Here we need the completeness assumption on E which implies either $E \vdash \phi_i$ or $E \vdash \neg\, \phi_i$. In the first case, $\phi_i \in Thm(E)$, we get $f(i) \in A$ and $\{p_{f(i)}\}\, S_{f(i)}\, \{q_{f(i)}\} \in PC(\Sigma,E)$ by Lemma 4.2. Thus, $\{\phi_i \wedge p_{f(i)}\}\, S_{f(i)}\, \{q_{f(i)}\} \in PC(\Sigma,E) \subset PC(PA(\Sigma,E))$.

In the second case, $\neg\, \phi_i \in Thm(E)$, we get $\{\phi_i \wedge p_{f(i)}\}S_{f(i)}\{q_{f(i)}\} \in PC(PA(\Sigma,E))$ trivially. $\square$

Certainly, there are structures A possessing a complete logic for partial correctness and for which the standard Hoare logic is *not* complete (see [4]. But what comes to mind, in addition to applications, is the following.

4.4 QUESTION. *Is the statement of Theorem* 4.1 *true without the hypothesis of completeness on the specification*?

An affirmative answer, which we think obtains, would be a strong statement about the genericity of Hoare's logic.

REFERENCES

[1] APT, K.R., *Ten years of Hoare's logic*, a survey in F.V. Jensen, B.H. Mayoh and K.K. Møller (eds), Proceedings from 5th Scandinavian Logic Symposium, Aalborg University Press, Aalborg, 1979, 1-44. (A second edition of this paper will appear in ACM Transactions on Programming Languages and Systems).

[2] DE BAKKER, J.W., *Mathematical theory of program correctness*, Prentice-Hall International, London, 1980.

[3] BERGSTRA, J.A., J. TIURYN & J.V. TUCKER, *Floyd's principle, correctness theories and program equivalence*, Mathematical Centre, Department of Computer Science Research Report IW 145, Amsterdam, 1980. (To appear in Theoretical Computer Science.)

[4] BERGSTRA, J.A., A. CHMIELINSKA & J. TIURYN, *Hoare's logic is incomplete when it does not have to be*, forthcoming.

[5] BERGSTRA, J.A. & J.V. TUCKER, *Some natural structures which fail to possess a sound and decidable Hoare-like logic for their while-programs* (To appear in Theoretical Computer Science. An earlier edition of this paper is registered at the Mathematical Centre as report IW 136/80).

[6] BERGSTRA, J.A. & J.V. TUCKER, *Algebraically specified programming systems and Hoare's logic*, Mathematical Centre, Department of Computer Science Research Report IW 143, Amsterdam, 1980.

[7] BERGSTRA, J.A. & J.V. TUCKER, *Expressiveness and the completeness of Hoare's logic*, Mathematical Centre, Department of Computer Science Research Report IW 149, Amsterdam, 1980.

[8] BERGSTRA, J.A. & J.V. TUCKER, *On the refinement of specifications and Hoare's logic*, Mathematical Centre, Department of Computer Science Research Report IW 155, Amsterdam, 1980.

[9] BERGSTRA, J.A. & J.V. TUCKER, Hoare's logic and Peano's arithmetic, Mathematical Centre, Department of Computer Science Research Report IW 160, Amsterdam, 1981.

[10] COOK, S.A., *Soundness and completeness of an axiom system for program verification*, Siam J. Computing 7 (1978) 70-90.

[11] GOGUEN, J.A., J.W. THATCHER & E.G. WAGNER, *An initial algebra approach to the specification, correctness and implementation of abstract data types*, in R.T. Yeh (ed.), *Current trends in programming methodology* IV, *Data structuring*, Prentice-Hall, Engelwood Cliffs, New Jersey, 1978, 80-149.

[12] GREIBACH, S.A., *Theory of program structures: schemes, semantics, verification*, Springer-Verlag, Berlin, 1975.

[13] HOARE, C.A.R., *An axiomatic basis for computer programming*, Communications Association Computing Machinery 12 (1969) 576-580.