A PROPOSITIONAL VERSION OF HOARE'S LOGIC

by

J.A. Bergstra* and J. Terlouw**

ABSTRACT: We describe a propositional version of Hoare's Logic and prove it to be complete.

KEYWORDS AND PHRASES: Partial correctness, while-programs, Hoare's Logic.

*) Dept. of Computer Science, University of Leiden, Wassenaarseweg 80, P.O. Box 9512, 2300 RA Leiden, The Netherlands.

**) Mathematical Institute, University of Utrecht, Budapestlaan 6, P.O. box 80.101, 2508 TA Utrecht, The Netherlands.

1. INTRODUCTION

Hoare's logic is a proof system for deriving partial correctness conditions concerning programs working on a structure. Typically partial correctness information about larger (composed) programs is derived using such information about smaller programs (subprograms). In this sense it is closely related to denotational semantics. Usually atomic or primitive programs are just assignments and information is obtained using the assignment axiom:

$$\{p[x/\tau]\} \quad x := \tau \{p\} .$$

This slightly obscures the fact that HL is perfectly suited for working from arbitrary assumptions of the form

$$\{p\}\ A\ \{q\}$$

with A an atomic action symbol. (Of course this aspect is generally used in proof rules for recursive procedures).

We will define a propositional system, in which no variables and, a fortiori, no assignments occur, and where all information on atomic action symbols must be explicitly specified. After a brief introduction to syntax and semantics of this system a Hoare logic for it is presented and shown to be sound and complete.

Some discussion of the relevance of this completeness proof seems in place here. We restrict ourselves to while-prgrams. For such programs, when working on algebraic structures and with all and only assignments as atomic actions the situation is reasonably well understood. First of all HL is then relative complete in the sense of COOK [4] . This means that if $\mathcal{A}$ is an expressive Σ structure for some signature Σ then $HL(\Sigma, Th(\mathcal{A}))$ is complete, (here $Th(\mathcal{A})$ is the full oracle for $\mathcal{A}$). Now from BERGSTRA & TUCKER [2] it follows that $HL(\Sigma, Th(\mathcal{A}))$ may be complete also if $\mathcal{A}$ is not expressive; further in BERGSTRA & TUCKER [3] a canonical construction is given of incom-

plete specifications (Σ,T), with $HL(\Sigma,T)$ still complete in a meaningful way. ((Σ,T) is incomplete if it is not of the form $(\Sigma,Th(\mathcal{A}))$ for some $\mathcal{A} \in ALG(\Sigma)$). We conclude that relative completeness explains only partially to what extent HL is complete. Especially it does not answer the following question: are there any rules similar in spirit to those of HL, as universally valid, but nonderivable from the original ones. Clearly, if one allows to fix a signature Σ, or rather, a specification (Σ,E) then WAND [5] presents an incompleteness of HL which can be seen as a new rule or axiom. To add something that specific to HL, however, seems improper.

Our completeness result indicates an unrestricted context where HL is complete, and where consequently no new valid rules can be found. We leave, at present, unanswered the next question: where begins HL to be incomplete ?
Having fixed a syntax Σ, a semantics $\models$ and a proofsystem $\vdash$ our completeness result reads as follows:

THEOREM. Let Γ be a finite set of assumptions, $\{p\}S\{q\}$ be an asserted *while*-program over Σ then $\Gamma \models \{p\}S\{q\}$ if, and only if $\Gamma \vdash \{p\}S\{q\}$.

Assumptions can have two forms: p, with p a Σ- assertion or $\{p\}A\{q\}$ an asserted atomic program over Σ. If Γ would be allowed to be infinite then $\Gamma \models \{p\}S\{q\}$ need not imply the existence of a finite subset Γ' of Γ such that $\Gamma' \models \{p\}S\{Q\}$. Proofs being finite and sound this implies that the finiteness condition is essential, thus establishing a clear difference with the situation in pure propositional calculus. As an example consider:

$$\Gamma = \{\{p_i\}\ A\ \{p_{i+1}\} \mid i \in \omega\} \cup \{p_i \wedge \neg b \rightarrow q \mid i \in \omega\}$$

Then $\Gamma \models \{p_0\}$ *while* b *do* A *od* $\{q\}$, but no finite subset Γ' of Γ does the same.

The above theorem is shown for deterministic *while*-programs. A natural generalisation of the result to a language containing recursion is not so easy to find. Various kinds of concurrency offer no essential difficulties however. For a treatment of program equivalence like in BERGSTRA & TERLOUW [1] various nontrivial changes of the system are required.

2.1. *Syntax*. The signature Σ contains
names $A_0, A_1, \ldots$ for atomic action symbols,
names $b_0, b_1, \ldots$ for boolean atoms and
names $p_0, p_1, \ldots$ for primitive assertions.

Composite boolean expressions are generated by means of the connectives $\neg$ and $\vee$, starting on the boolean atoms b_i. Similarly (composite) assertions are generated from boolean atoms plus primitive assertions.

The A_i are atomic programs, if B is a boolean and S_0, S_1 are programs then so are:

$S_0 ; S_1$

$\underline{\text{if}}$ B $\underline{\text{then}}$ S_0 $\underline{\text{else}}$ S_1 $\underline{\text{fi}}$

$\underline{\text{while}}$ B $\underline{\text{do}}$ S_0 $\underline{\text{od}}$

A partial correctness condition (or:asserted program) is a construct of the form $\{p\}S\{q\}$ with p, q assertions and S a program.

The following abbrevations will be used:
$P \wedge Q \equiv \neg(\neg P \vee \neg Q)$, $P \to Q \equiv \neg P \vee Q$,
$P \leftrightarrow Q \equiv (P \to Q) \wedge (Q \to P)$,
$T \equiv b_0 \vee \neg b_0$, $F \equiv \neg T$.

2.2. $\underline{\text{Semantics}}$. A structure $\mathcal{A}$ is a domain V together with: an interpretation $b^{\mathcal{A}} \subseteq V$ for each boolean atom b, an interpretation $p^{\mathcal{A}} \subseteq V$ for each primitive assertion p and, for each atomic action symbol A an interpretation $A^{\mathcal{A}} \subseteq V \times V$. $A^{\mathcal{A}}$ must be a single valued relation, thus A is interpreted as a partial $\underline{\text{deterministic}}$ operation on the state space V.
Then for states $s \in V$ and assertions P the satisfaction relation $\mathcal{A}, s \models P$ is inductively defined as follows.

(i) $\mathcal{A}, s \models b_i$ iff $s \in b_i^{\mathcal{A}}$
$\mathcal{A}, s \models p_i$ iff $s \in p_i^{\mathcal{A}}$

(ii) $\mathcal{A}, s \models P \vee Q$ iff $\mathcal{A}, s \models P$ or $\mathcal{A}, s \models Q$
$\mathcal{A}, s \models \neg P$ iff $\mathcal{A}, s \not\models P$.

Further for every program S a single valued binary relation $M_{\mathcal{A}}(S) \subseteq V \times V$ is defined by:

(i) $M_{\mathcal{A}}(A_i) = A_i^{\mathcal{A}}$

(ii) $M_{\mathcal{A}}(S_0; S_1) = M_{\mathcal{A}}(S_0) \circ M_{\mathcal{A}}(S_1)$

(iii) $M_{\mathcal{A}}(\underline{\text{if}}$ B $\underline{\text{then}}$ S_0 $\underline{\text{else}}$ S_1 $\underline{\text{fi}}) =$

$\{(s,t) \in V \times V \mid$ if $\mathcal{A},s \models B$ then $(s,t) \in M_{\mathcal{A}}(S_0)$
else $(s,t) \in M_{\mathcal{A}}(S_1)\}$

(iv) $M_{\mathcal{A}}(\underline{\text{while}}$ B $\underline{\text{do}}$ S $\underline{\text{od}}) = \{(s,t) \in V \times V \mid$ there are $s_0,\ldots,s_n \in V$ such that $s_0 = s_1$, $s_n = t$ and $\forall_{i<n}\, \mathcal{A}, s_i \models B$ and $(s_i,s_{i+1}) \in M_{\mathcal{A}}(S)$ and finally: $\mathcal{A}, t \models \neg B\}$.

Finally we define:

$\mathcal{A} \models P$ iff for all $s \in V$ $\mathcal{A}, s \models P$ and

$\mathcal{A} \models \{P\}S\{Q\}$ iff for all $s,t \in V$ with $\mathcal{A}, s \models P$ and $(s,t) \in M_{\mathcal{A}}(S)$ also $\mathcal{A}, t \models q$

In the sequel Γ will always represent a finite set of assertions P and/or asserted atomic programs $\{p\}A_i\{q\}$. We split up Γ in $\Gamma^{SP} \cup \Gamma^{PR}$ where Γ^{SP} contains the assertions (specification part) and Γ^{PR} contains the asserted programs (program part). Then by definition $\Gamma \models \{p\}S\{q\}$ iff for all $\mathcal{A}$ with $\mathcal{A} \models \Gamma$, $\mathcal{A} \models \{p\}S\{q\}$ as well.

2.3. Nondeterministic semantics. A straightforward generalisation of the previous semantics results if we allow the $M_{\mathcal{A}}(A_i)$ to be multiple valued relations. In this case programs will be nondeterministic. For nondeterministic structures $\mathcal{A}$, the relations $\mathcal{A}, s \models P$, $\mathcal{A} \models P$, and $\mathcal{A} \models \{p\}S\{q\}$ have exactly the same definitions. We write $\Gamma \models_N \{p\}S\{q\}$ to indicate that in all nondeterministic (including of course the deterministic ones) Γ-structures $\mathcal{A}$, $\mathcal{A} \models \{p\}S\{q\}$. The following lemma will prove to be quite useful in the sequel.

2.4. Lemma. $\Gamma \models \{P\}S\{q\} \Leftrightarrow \Gamma \models_N \{p\}S\{q\}$.

Proof: „$\Leftarrow$" is trivial because all deterministic Γ structures are covered by $\Gamma \models_N \{p\}S\{q\}$ as well.

For „$\Rightarrow$" we introduce a transformation $\mathcal{A} \to \mathcal{A}'$ that transforms nondeterministic structures into deterministic ones with the following condition satisfied: for all $\{p\}S\{q\}$

$$\mathcal{A} \models \{p\}S\{q\} \Leftrightarrow \mathcal{A}' \models \{p\}S\{q\}.$$

Given this „$\Rightarrow$" follows by contraposition. Indeed, if $\mathcal{A} \not\models \{p\}S\{q\}$, $\mathcal{A} \models \Gamma$ then $\mathcal{A}'$ has the same properties thus $\Gamma \not\models \{p\}S\{q\}$.

We will now describe the transformation. Let $\mathcal{A}$ have state space V, then the domain of $\mathcal{A}'$ is $V' = \bigcup_{n \geq 1} V^n$. Interpretations are as follows:

$$\mathcal{A}', (s_0,\ldots,s_k) \models b_i \quad \text{iff} \quad \mathcal{A}, s_0 \models b_i$$

$$\mathcal{A}', (s_0,\ldots,s_k) \models p_i \quad \text{iff} \quad \mathcal{A}, s_0 \models p_i$$

$$M_{\mathcal{A}'}(A_i) = \{((s_0,s_1,\ldots,s_{k+1}), (s_1,\ldots,s_{k+1}) \mid (s_0,s_1) \in M_{\mathcal{A}}(A_i)\}$$

Then using induction on the complexity of assertions and programs one finds:

(i) $\mathcal{A}', (s_0,\ldots,s_k) \models P \Leftrightarrow \mathcal{A}, s_0 \models P$, and

(ii) $((s_0,\ldots,s_k), (t_0,\ldots,t_\ell) \in M_{\mathcal{A}}(s)$ iff

$t_0 = s_{k-\ell} \wedge \ldots \wedge t_\ell = s_k$ and $(s_0, t_0) \in M_{\mathcal{A}}(S)$.

Further, if $(S_0,\ldots,S_k)$ is a computation sequence of S in $\mathcal{A}$ then so is $(S_0,\ldots,S_k)$, $(S_1,\ldots,S_k),\ldots,(S_k)$ in $\mathcal{A}'$.

It is obvious that $\mathcal{A}'$ is a deterministic structure. From (i) and (ii) it follows that $\mathcal{A} \models \{p\}S\{q\}$ implies $\mathcal{A}' \models \{p\}S\{q\}$. For the converse implication assume $\mathcal{A} \not\models \{p\}S\{q\}$ then there must be a computation sequence, leading in $\mathcal{A}$ from p via S to $\neg q$, but then this phenomenon is reflected in $\mathcal{A}'$.

3.1. A Proof System

Axioms: $\{T\}\ A_i\ \{T\}$, $\{F\}\ A_i\ \{F\}$

Rules:
$$\frac{\{P_1\}\ A_i\ \{Q_1\}\ \{P_2\}\ A_i\ \{Q_2\}}{\{P_1 \wedge P_2\}\ A_i\ \{Q_1 \wedge Q_2\}} \qquad \text{(conjunction rule)}$$

$$\frac{\{P_1\}\ A_i\ \{Q_1\}, \{P_2\}\ A_i\ \{Q_2\}}{\{P_1 \vee P_2\}\ A_i\ \{Q_1 \vee Q_2\}} \qquad \text{(disjunction rule)}$$

$$\frac{\Gamma^{SP} \models P \to P',\ \{P'\}\ S\ \{Q'\},\ \Gamma^{SP} \models Q' \to Q}{\{P\}\ S\ \{Q\}} \qquad \text{(rule of consequence)}$$

$$\frac{\{P\}\ S_0\ \{R\}, \{R\}\ S_1\ \{Q\}}{\{P\}\ S_0\ ;\ S_1\ \{Q\}} \qquad \text{(composition rule)}$$

$$\frac{\{P \wedge B\}\ S_0\ \{Q\},\ \{P \wedge \neg B\}\ S_2\ \{Q\}}{\{P\}\ \underline{\text{if}}\ B\ \underline{\text{then}}\ S_0\ \underline{\text{else}}\ S_1\ \underline{\text{fi}}\ \{Q\}} \qquad \text{(conditional rule)}$$

$$\frac{\{P \wedge B\}\ S\ \{P\}}{\{P\}\ \underline{\text{while}}\ B\ \underline{\text{do}}\ S\ \underline{\text{od}}\ \{P \wedge \neg B\}} \qquad \text{(iteration rule)}$$

This proof system constitutes an inductive definition of a relation $\Gamma \vdash \{p\}\ S\ \{q\}$, where it is of course understood that if $\{p\}\ A_i\ \{q\} \in \Gamma^{PR}$, then $\Gamma \vdash \{p\}\ A_i\ \{q\}$.

Observe that our system contains exactly the standard rules plus some extra rules for atomic programs all of which would be derived rules if atomic actions are assignments and the assignment axiom is assumed. We will then prove the completeness of this system.

3.2. THEOREM. For any finite Γ

$$\Gamma \models \{p\}S\{q\} \Leftrightarrow \Gamma \vdash \{p\}S\{q\}$$

PROOF. By lemma 2.4 it suffices to prove

$$\Gamma \models_N \{p\}S\{q\} \Leftrightarrow \Gamma \vdash \{p\}S\{q\}$$

Here the proof of „⇐" (soundness) is a routine induction on proof lengths based on the observation that the individual rules are sound for the non-deterministic semantics. The work lies in the proof of

$$\Gamma \models_N \{p\}S\{q\} \Rightarrow \Gamma \vdash \{p\}S\{q\}.$$

As a first step choose k to be a sufficiently large number so that $\Sigma_k = \{b_0,\ldots,b_k,p_0,\ldots,p_k,A_0,\ldots,A_k\}$ contains all atomic symbols that occur in Γ and in $\{P\}\ S\ \{q\}$. Let $\models^k_N$ be semantic entailment relative to all Σ_k structures.

Assuming $\Gamma \models_N \{p\}\ S\ \{q\}$ we derive $\Gamma \models^k_N \{p\}\ S\ \{q\}$ because a Σ_k structure $\mathcal{A}$ with $\mathcal{A} \models \Gamma$ and $\mathcal{A} \not\models \{p\}S\{q\}$ can be trivially expanded to a Σ structure having the same properties. We then focus on proving for all $\{p\}S\{q\}$ over Σ_k the implication

$$\Gamma \models^k_N \{p\}S\{q\} \Rightarrow \Gamma \vdash \{p\}S\{q\}.$$

This proof in turn rests upon the construction of a Σ_k structure $\mathcal{A}_\Gamma$ with $\mathcal{A}_\Gamma = \Gamma$ and endowed with the following remarkable property

$$\mathcal{A}_\Gamma \models \{p\}S\{q\} \Rightarrow \Gamma \vdash \{p\}S\{q\}.$$

3.3 In this section of the proof we concentrate on the construction of $\mathcal{A}_\Gamma$.

Let D^k be the set of all formulas

$$b'_0 \wedge \ldots \wedge b'_k \wedge p'_0 \wedge \ldots \wedge p'_k$$

where for each i $(0 \le i \le k)$, b'_i, (p'_i) is b_i or $\neg b_i$ (p_i or $\neg p_i$). Clearly D^k has cardinality 2^{2k+2}.

Now take D_Γ to be the subset of D^k containing those formulas $s \in D^k$ that satisfy $s \models \Gamma^{SP}$. Interpretations of boolean atoms and primitive assertions are given by

$$\mathcal{A}_\Gamma, s \models b_i \text{ iff } s \models b_i$$

$$\mathcal{A}_\Gamma, s \models p_i \text{ iff } s \models p_i$$

It remains to fix an interpretation for the atomic action symbols A_i. It is at this stage worthwhile to notice that taking $M(A_i) = \emptyset$ already leads to a Γ structure, and that if two meaning functions M_1 and M_2 on D_Γ turn it into a Γ structure, then also the meaning function M defined by $M(A_i) = M_1(A_i) \cup M_2(A_i)$ leads to a Γ structure. It is therefore possible to obtain a maximal M ($M_{\mathcal{A}_\Gamma}$) as follows:

$(s,t) \in M(A_i) \Leftrightarrow$ whenever $\{p\}A_i\{q\} \in \Gamma$ then $[s \models P \Rightarrow t \models q]$.

3.4. It remains to show that $\mathcal{A}_\Gamma \models \{p\}S\{q\}$ implies $\Gamma \vdash \{p\}S\{q\}$, which is of course done via induction on the structure of S. The hardest case is that S is atomic. So assume $\mathcal{A}_\Gamma \models \{p\}A_i\{q\}$. There are two cases, if Γ contains no assumption of the form $\{R\}A_i\{R'\}$ then $M(A_i) = D_\Gamma \times D_\Gamma$. Thus $D_\Gamma \models q$, but then $\Gamma^{SP} \models q$, and $\Gamma^{SP} \models T \to q$. Of course $\Gamma^{SP} \models p \to T$ and thus applying the axiom $\{T\}A_i\{T\}$ and the rule of consequence one obtains $\Gamma \vdash \{p\}A_i\{q\}$.

In the second case observe that $\Gamma^{SP} \models \bigvee_{s \in D_\Gamma} s$, and therefore due to the disjunction rule it suffices to prove $\Gamma \vdash \{p \wedge S\}A_i\{q\}$ for all $s \in D_\Gamma$. This reduces our problem to the implication

$$\Gamma \models \{s\}A_i\{q\} \Rightarrow \Gamma \vdash \{s\}A_i\{q\}.$$

(In the cases that $\Gamma \models s \to \neg p$ the $\{F\}A_i\{F\}$ axiom and the rule of consequence do the work.)

Let $R_s = \{\bigwedge R' \mid$ for some R $s \models R$ and $\{R\}A_i\{R'\} \in \Gamma\}$

Observe that $(s,t) \in M(\mathcal{A})$ implies $t \models R_s$ and conversely because of the maximality of M. Therefore $\mathcal{A}_\Gamma \models R_s \to q$, thus $\Gamma^{SP} \models R_s \to q$.

Applying the rule of consequence and conjunction rule several times one finds $\Gamma \vdash \{s\}A_i\{R_s\}$.

Applying the rule of consequence once more gives $\Gamma \vdash \{s\}A_i\{q\}$. This is sufficient to show $\Gamma \vdash \{p\}A_i\{q\}$ as required.

Next we consider the case $S \equiv$ *if* B *then* S_0 *else* S_1 *fi*.

Clearly $\mathcal{A}_\Gamma \models \{p\}S\{q\}$ implies $\mathcal{A}_\Gamma \models \{p \wedge B\}S_0\{q\}$ and $\mathcal{A}_\Gamma \models \{p \wedge \neg B\}S_1\{q\}$.

According to the induction hypothesis $\Gamma \vdash \{p \wedge B\}S_0\{q\}$ and $\Gamma \vdash \{p \wedge \neg B\}S_1\{q\}$ which, using the conditional rule combines into the required $\Gamma \vdash \{p\}S\{q\}$.

Then let $S \equiv S_0;S_1$ and $\mathcal{A}_\Gamma \models \{p\}S\{q\}$.

Let $R \equiv W\{t \in D_\Gamma \mid \exists s \in D_\Gamma,\ s \models p \ \&\ (s,t) \in M(S_0)\}$.

Clearly $\mathcal{A}_\Gamma \models \{p\}S_0\{R\}$ and $\mathcal{A}_\Gamma \models \{R\}S_1\{q\}$.

Thus, using the induction hypothesis twice and the composition rule there after one obtains $\Gamma \vdash \{p\}S\{q\}$.

Finally let $S \equiv$ *while* B *do* S_0 *od* and assume $\mathcal{A}_\Gamma \models \{p\}S\{q\}$.

Let $INV(p,B,S_0)$ be the smallest subset I of D_Γ that satisfies the following conditions:

(i) if $s \models p$ then $s \in I$

(ii) if $s \in I$, $s \models B$ and $(s,t) \in M_{\mathcal{A}_\Gamma}(S_0)$ then $t \in I$.

Let $INV \equiv W\{s \in INV(p,B,S_0)\}$. Observe: $\mathcal{A}_\Gamma \models p \rightarrow INV$, $\mathcal{A}_\Gamma \models \{INV \wedge B\}S_0\{INV\}$ and $\mathcal{A}_\Gamma \models (INV \wedge \neg B) \rightarrow q$. Then, applying the rule of consequence on the result of the induction hypothesis gives the required result:

$\Gamma \vdash \{p\}S\{q\}$.

REFERENCES.

[1] BERGSTRA J.A. & J. TERLOUW, A characterization of program equivalence in terms of Hoare's logic, Leiden report no. 81-10.

[2] BERGSTRA J.A. & J.V. TUCKER, Expressiveness and the completeness of Hoare's Logic. Mathematical Centre Research Report IW 149/80 Amsterdam 1980.

[3] BERGSTRA J.A. & J.V. TUCKER, Two theorems on the completeness of Hoare's Logic. Mathematical Centre Report IW 16?/81 Amsterdam 1981.

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

[5] WAND M., A new incompleteness result for Hoare's system, J.A.C.M. 25 (1978) 168-175.