Properties of models which are complete for Hoare logic

R. Hoofman

RUU-CS-90-1
January 1990

Utrecht University

Department of Computer Science

Padualaan 14, P.O. Box 80.089,
3508 TB Utrecht, The Netherlands,
Tel. : ... + 31 - 30 - 531454

Properties of models which are complete for Hoare logic

R. Hoofman

Technical Report RUU-CS-90-1
January 1990

Department of Computer Science
Utrecht University
P.O.Box 80.089
3508 TB Utrecht
The Netherlands

Properties of Models which are Complete for Hoare Logic

[illegible]
Department of Computer Science, University of Utrecht
P.O. Box 80.089, 3508 TB Utrecht, the Netherlands

January 5, 199[illegible]

Abstract

A model M is said to be complete for Hoare logic iff Hoare logic is sound and complete with respect to M. A model M is said to be expressive iff for each program α and each formula ϕ the weakest precondition $wp(\alpha, \phi)$ is definable, i.e. there is a formula which has $wp(\alpha, \phi)$ as interpretation. It is known that all expressive models are complete, and that complete models exist which are not expressive. We show that all complete models have a property which comes very close to expressiveness: with respect to any complete model M, there always is a definable set between the total weakest precondition and the weakest precondition. This property is called [illegible]. Among other things it follows that weakest preconditions of total programs are always definable in a complete model.

1 Introduction

In this section we introduce Hoare logic. More details can be found in [illegible]. Let $\mathcal{L}$ be a first order language, with a basis consisting of certain function and predicate symbols. Let $x, y, z, \ldots$ denote variables in this language, $t, \ldots$ terms, [illegible] formulae, and $b, \ldots$ quantifier-free formulae. Define a programming language $\mathcal{P}$ with statements α by the following syntax:

$$\alpha ::= x := t \mid \alpha;\alpha \mid \text{if } b \text{ then } \alpha \text{ else } \alpha \text{ fi} \mid \text{while } b \text{ do } \alpha \text{ od}$$

For clarity we write sometimes statements enclosed in square brackets. $\mathcal{P}$ is the well-known language of while programs. Define a Hoare logic $\mathcal{HL}$ over $\mathcal{L}$ with formulae as follows:

$$[illegible] \mid [\alpha]\phi(\psi)$$

ISSN:0924-3275

The deduction system $\mathcal{D}_0$ for $\mathcal{HL}$ consists of the usual while-rule, the rules for composition, conditionals and consequence, and the assignement axiom.
Let M be a (set-theoretic) model of $\mathcal{L}$. Let Σ be the set of states, i.e. the set of functions σ from the set of variables to the domain of M. Then the interpretation of ϕ in M (denoted by $M[\phi]$) is a subset of Σ in the usual way. In particular $M[false] = \emptyset$ and $M[true] = \Sigma$.

Definition 1 *A subset $S \subseteq \Sigma$ is* definable *iff there is a formula ϕ such that $M[\phi] = S$. In this case we say that ϕ* defines *S.*

Suppose further that M gives a semantics to $\mathcal{P}$, i.e. we can interprete statements α as functions $M[\alpha]$ from Σ to Σ. Among other things we require the semantics to satisfy the following property:

$$\text{If } x \text{ does not occur in } \alpha, \text{ then} \forall\sigma\forall d : M[\alpha](\sigma[d/x]) = (M[\alpha](\sigma))[d/x] \qquad (1)$$

where d is an element of the domain of M, and $\sigma[d/x]$ denotes the function equal to σ except that it delivers d as value for x. Further all the statements should have the usual effect on states, e.g. assignement should correspond to substitution in that $M[x := t](\sigma) = \sigma[M[t](\sigma)/x]$.

Definition 2 *Let α be a program, and $S \subseteq \Sigma$.*

1. *The* weakest precondition *of α and S (denoted by $wp(\alpha, S)$) is defined as $\{\sigma \in \Sigma | \forall\sigma'(M[\alpha](\sigma) = \sigma' \Rightarrow \sigma' \in S)\}$.*

2. *The* total weakest precondition *of α and S (denoted by $twp(\alpha, S)$) is defined as $\{\sigma \in \Sigma | \exists\sigma'(M[\alpha](\sigma) = \sigma' \wedge \sigma' \in S)\}$.*

For formulae ϕ we abbreviate $wp(\alpha, M[\phi])$, resp. $twp(\alpha, M[\phi])$ as $wp(\alpha, \phi)$, resp. $twp(\alpha, \phi)$.

Lemma 3 *For all programs α and $S \subseteq \Sigma$ we have $wp(\alpha, S) = twp(\alpha, S) \cup wp(\alpha, false)$.*

We can view models M of $\mathcal{L}$ which give a semantics to $\mathcal{P}$ as models of $\mathcal{HL}$ in the usual way. We write $\models_M l$ iff l is true in M. Let $\mathcal{D}_M$ be the deduction system consisting of $\mathcal{D}_0$, with as additional axioms all ϕ such that $\models_M \phi$. We write $\vdash_M l$ iff l is deducible in $\mathcal{D}_M$.

Definition 4 *M is called* complete *iff $\forall l(\models_M l \Rightarrow \vdash_M l)$.*

Each model M of $\mathcal{HL}$ satisfies the reverse of definition 4, i.e. $\forall l(\vdash_M l \Rightarrow \models_M)$. This is the *soundness* of Hoare logic.
However there are a lot of models which are not complete. Let $\mathcal{M}_\mathcal{L}$ (denoted simply by $\mathcal{M}$) be the class of all complete models of language $\mathcal{L}$. A subclass of M is the class of expressive models.

Definition 5 *M is* expressive *iff for every α and for every ϕ the set $wp(\alpha, \phi)$ is definable.*

Let $\mathcal{E}_{\mathcal{L}}$ (denoted simply by $\mathcal{E}$) be the class of expressive models. It is well-known that $\mathcal{E} \subseteq \mathcal{M}$, i.e. every expressive model is complete. In fact this inclusion is strict.

Theorem 6 *([1]) $\mathcal{E} \subset \mathcal{M}$*

Proof: Let $\mathcal{L}$ be the language of Peano arithmetic, and N the standard model of the natural numbers. Then $N \in \mathcal{M}$ ([2]). Let M be a non-standard model with the same theory. It follows that the same formulae of $\mathcal{HL}$ are true in both models, hence M is complete. Take

$$\alpha = [\text{while } x \neq 0 \text{ do } x := x - 1 \text{ od}]$$

then $wp(\alpha, false) = \{\sigma | \sigma(x) \text{is non-standard}\}$ in M. It is well-known that this set is not definable, hence M is not expressive. ■

In this article we shall prove that although complete models need not be expressive, they satisfy a property which comes very close to it, called *weak expressivity.* This means that for each α and ϕ there is a definable set between $twp(\alpha, \phi)$ and $wp(\alpha, \phi)$. As a consequence we find among other things that weakest preconditions of total programs are always definable in a complete model.
In the next section we will define when a model is weak expressive, and we will give some theorems and lemmas about weak expressive models. In the last section we prove a certain basic theorem, and use that to show that complete models are weak expressive.

2 Weak Expressivity

In this section we will define *weak* expressive models, and consider some properties of them.

Definition 7 *A model M is* weak expressive *iff for every α and every ϕ there is a ψ such that*

$$twp(\alpha, \phi) \subseteq M[\psi] \subseteq wp(\alpha, \phi)$$

(In fact we consider only pairs (α, ϕ) such that the variables in α are different from the bound variables in ϕ, but we can always rename.)

So although $R = wp(\alpha, \phi)$ need not be definable in a weak expressive model, there always is a definable set S, which differs from R only in that it does not contain all elements on which α does not terminate. It is easy to see that every expressive model is weak expressive.
Weak expressive models have some interesting properties.

Theorem 8 *If M is weak expressive, $\models_M \phi \Rightarrow \psi$, and $wp(\alpha, \phi)$ is definable, then $wp(\alpha, \psi)$ is definable.*

Proof: Suppose ϕ' defines $wp(\alpha, \phi)$. There is a ψ' such that $twp(\alpha, \psi) \subseteq M[\psi'] \subseteq wp(\alpha, \psi)$. We will show that $\phi' \vee \psi'$ defines $wp(\alpha, \psi)$.

- $M[\phi' \vee \psi'] = M[\phi'] \cup M[\psi'] = wp(\alpha, \phi) \cup M[\psi']$ *subseteq* $wp(\alpha, \psi) \cup M[\psi'] \subseteq wp(\alpha, \psi) \cup wp(\alpha, \psi) = wp(\alpha, \psi)$
- $wp(\alpha, \psi) = twp(\alpha, \psi) \cup wp(\alpha, false) \subseteq M[\psi'] \cup wp(\alpha, false) \subseteq M[\psi'] \cup wp(\alpha, \phi) = M[\psi'] \cup M[\phi'] = M[\psi' \vee \phi']$

■

Theorem 9 *If M is weak expressive, then the following are equivalent for every α:*

- *$wp(\alpha, false)$ is definable*
- *$wp(\alpha, \phi)$ is definable for all ϕ*

Proof: By theorem 8. ■

Theorem 10 *If M is weak expressive, then the following are equivalent:*

- *M is expressive*
- *$wp(\alpha, false)$ is definable for all α*

Proof: By theorem 9. ■

Theorem 11 *If M is weak expressive, then all weakest preconditions of total programs are definable.*

Proof: If α total, then $wp(\alpha, false) = false$. The result now follows by theorem 9. ■

In the next section we will prove that a model is weak expressive iff it is "weak expressive for every program and every formula without quantifiers". To prepare for it we give some definitions and lemmas.

Definition 12 *The* projection *of a set $S \subseteq \Sigma$ on the x-component is the set $\{\sigma \in \Sigma | \exists d : \sigma[d/x] \in S\}$. Notation: $\exists x S$.*
The complement *of a set $S \subseteq \Sigma$ is the set $\{\sigma \in \Sigma | \sigma \notin S\}$. Notation: $\neg S$.*

It is clear that $M[\exists x\phi] = \exists x M[\phi]$, and $M[\neg\phi] = \neg M[\phi]$.

Definition 13 *Let $R, S \subseteq \Sigma$. R is a* reduct *of S ($R \ll S$) iff one of the following clauses holds:*

- $R = \exists x R'$, *and* $R' \ll S$
- $R = \neg R'$, *and* $R' \ll S$
- $R{=}S$

Let ϕ^1 denote a prenex normal form (i.e. ϕ^1 is a formula of L with all quantifiers in front) for each formula ϕ. Let ϕ^2 be the equivalent formula obtained by changing all $\forall x$ in $\neg\exists x\neg$. Finally let ϕ^0 be the remainder of ϕ^2 when we remove all $\exists x$ and $\neg$ in front. We now have $M[\phi] \ll M[\phi^0]$, and ϕ^0 is quantifier-free.

Definition 14 *A* prefix π *is a finite sequence of symbols $\exists x$ and $\neg$. A prefix is* even *iff it contains an even number of $\neg$-symbols, otherwise it is* odd. *The* length *of a prefix π is the number of $\exists x$ and $\neg$ symbols it contains. Let $S \subseteq \Sigma$, then $\pi \circ S \subseteq \Sigma$ is defined as follows:*

- $\pi = \neg.\pi'$
  Then $\pi \circ S = \neg(\pi' \circ S)$
- $\pi = \exists x.\pi'$
  Then $\pi \circ S = \exists x(\pi' \circ S)$
- $\pi = \epsilon$
  Then $\pi \circ S = S$

By definition we have $R \ll S$ iff there exists a π such that $\pi \circ S = R$. If we define $\pi \circ \phi$ as the formula formed by concatenation of π and ϕ, then $M[\pi \circ \phi] = \pi \circ M[\phi]$.

Lemma 15 *Let $S \subseteq T$, π an arbitrary prefix. If π is even, then $\pi \circ S \subseteq \pi \circ T$, otherwise $\pi \circ T \subseteq \pi \circ S$.*

Proof: We prove the lemma by induction to the length n of π.

basis In the case $n = 0$ we have π is even, and $\pi \circ S = S \subseteq T = \pi \circ T$.

step Suppose the lemma is true for each prefix with length n, and let π be a prefix with length $n + 1$. We have the following cases:

- $\pi = \exists x.\pi'$
  Suppose π' even, then π is even. By induction hypothesis we have $\pi' \circ S \subseteq \pi' \circ T$. Suppose $\sigma \in \pi \circ S = \exists x(\pi' \circ S)$, then there is a d such that $\sigma[d/x] \in \pi' \circ S$. So $\sigma[d/x] \in \pi' \circ T$, and $\sigma \in \exists x(\pi' \circ T) = \pi \circ T$.
  Analogous if π' is odd.

- $\pi = \neg.\pi'$
  Suppose π' is even, hence π is odd. By the induction hypothesis $\pi' \circ S \subseteq \pi' \circ T$, so $\pi \circ T = \neg(\pi' \circ T) \subseteq \neg(\pi' \circ S) = \pi \circ S$.
  Analogous if π' is odd.

■

Lemma 16 *Let α be a program, π a prefix, $S \subseteq \Sigma$, and suppose the variables in α are different from those occuring in π. Then:*

1. *If π is even:*
   (a) $\pi \circ wp(\alpha, S) \subseteq wp(\alpha, \pi \circ S)$
   (b) $\pi \circ twp(\alpha, S) \supseteq twp(\alpha, \pi \circ S)$
2. *If π is odd:*
   (a) $\pi \circ twp(\alpha, S) \subseteq wp(\alpha, \pi \circ S)$
   (b) $\pi \circ wp(\alpha, S) \supseteq twp(\alpha, \pi \circ S)$

Proof: Simultaneously we prove these four statements by induction to the length n of π.

basis In the case $n = 0$, π is even, and we have $\pi \circ wp(\alpha, S) = wp(\alpha, S) = wp(\alpha, \pi \circ S)$. The same for *twp*.

hypothesis Suppose that if π has length n, then the theorem is true.

step Now suppose that π has length $n + 1$. We have to consider eight different cases. Here we write out two of them.

- We prove 1a) in the case that $\pi = \exists x.\pi'$ and π' even. By the induction hypothesis we have $\pi' \circ wp(\alpha, S) \subseteq wp(\alpha, \pi' \circ S)$. So:
  $\pi \circ wp(\alpha, S)$
  $=$
  $\exists x(\pi' \circ wp(\alpha, S))$
  $\subseteq$
  $\exists x(wp(\alpha, \pi' \circ S))$
  $=$
  $\exists x\{\sigma | \forall \sigma'(M[\alpha](\sigma) = \sigma' \Rightarrow \sigma' \in \pi' \circ S)\}$
  $=$
  $\{\sigma | \exists d \forall \sigma'(M[\alpha](\sigma[d/x]) = \sigma' \Rightarrow \sigma' \in \pi' \circ S)\}$
  = By property (1) of the semantics.
  $\{\sigma | \exists d \forall \sigma'(\exists \sigma''(M[\alpha](\sigma) = \sigma'' \wedge \sigma' = \sigma''[d/x]) \Rightarrow \sigma' \in \pi' \circ S)\}$

$=$
$\{\sigma|\exists d\forall\sigma''(M[\alpha](\sigma) = \sigma'' \Rightarrow \sigma''[d/x] \in \pi' \circ S)\}$
$\subseteq$
$\{\sigma|\forall\sigma''(M[\alpha](\sigma) = \sigma'' \Rightarrow \exists d(\sigma''[d/x] \in \pi' \circ S))\}$
$=$
$\{\sigma|\forall\sigma''(M[\alpha](\sigma) = \sigma'' \Rightarrow \sigma'' \in \exists x(\pi' \circ S))\}$
$=$
$wp(\alpha, \exists x(\pi' \circ S))$
$=$
$wp(\alpha, \pi \circ S)$

- We prove 2a) in the case that $\pi = \neg.\pi'$ and π' even. By the induction hypothesis we have $twp(\alpha, \pi' \circ S) \subseteq \pi' \circ twp(\alpha, S)$, so $\neg(\pi' \circ twp(\alpha, S)) \subseteq \neg twp(\alpha, \pi' \circ S)$. So:
$\pi \circ twp(\alpha, S)$
$=$
$\neg(\pi' \circ twp(\alpha, S))$
$\subseteq$
$\neg twp(\alpha, \pi' \circ S)$
$=$
$\neg\{\sigma|\exists\sigma'(M[\alpha](\sigma) = \sigma' \wedge \sigma' \in \pi' \circ S)\}$
$=$
$\{\sigma|\forall\sigma'(M[\alpha](\sigma) \neq \sigma' \vee \sigma' \notin \pi' \circ S)\}$
$=$
$\{\sigma|\forall\sigma'(M[\alpha](\sigma) = \sigma' \Rightarrow \sigma' \notin \pi' \circ S)\}$
$=$
$\{\sigma|\forall\sigma'(M[\alpha](\sigma) = \sigma' \Rightarrow \sigma' \in \neg(\pi' \circ S))\}$
$=$
$wp(\alpha, \neg(\pi' \circ S))$
$=$
$wp(\alpha, \pi \circ S)$

■

Lemma 17 *Let α be a program and ϕ, ψ formulae. Let ϕ^0 be the formula without quantifiers derived from ϕ as defined earlier, and let π be the prefix (so $\pi \circ \phi^0 \equiv \phi$). Let the sets of variables occuring in π and α be disjoint.*
Then

1. *If $twp(\alpha, \phi^0) \subseteq M[\psi]$, then*
   (a) *If π even, then $twp(\alpha, \phi) \subseteq M[\pi \circ \psi]$.*
   (b) *If π odd, then $M[\pi \circ \psi] \subseteq wp(\alpha, \phi)$.*

2. *If* $M[\psi] \subseteq wp(\alpha, \phi^0)$, *then*

   (a) *If* π *even, then* $M[\pi \circ \psi] \subseteq wp(\alpha, \phi)$.

   (b) *If* π *odd, then* $twp(\alpha, \phi) \subseteq M[\pi \circ \psi]$.

Proof: We shall only prove case 1a), as the remaining cases are similar.
We have $twp(\alpha, \phi^0) \subseteq M[\psi]$ and π even. By lemma 15 it follows that $\pi \circ twp(\alpha, \phi^0) \subseteq \pi \circ M[\psi]$. So with the help of lemma 16 case 1b) we get:
$twp(\alpha, \phi)$
$=$
$twp(\alpha, \pi \circ \phi^0)$
$\subseteq$
$\pi \circ twp(\alpha, \phi^0)$
$\subseteq$
$\pi \circ M[\psi]$
$=$
$M[\pi \circ \psi]$ ■

3 Complete models are weak expressive

Now we can prove the theorem about weak expressivity announced in the previous section. Using this theorem we shall prove that complete models are weak expressive.

Theorem 18 *A model* M *is weak expressive iff for every* α *and for every formula* b *without quantifiers there exists a formula* ψ *such that*

$$twp(\alpha, b) \subseteq M[\psi] \subseteq wp(\alpha, b)$$

Proof: The only if-part is trivial.
To prove the if-part suppose that α is a program and that ϕ is an arbitrary formula. We claim that there exists a ψ such that

$$twp(\alpha, \phi) \subseteq M[\psi] \subseteq wp(\alpha, \phi)$$

Write $\phi \equiv \pi \circ \phi^0$, with π a prefix and ϕ^0 quantifier-free. We assume that variables are suitably renamed such that ϕ and α have no variables in common. By assumption there is a ψ' such that

$$twp(\alpha, \phi^0) \subseteq M[\psi'] \subseteq wp(\alpha, \phi^0)$$

Now we have to consider two cases:

- π is even:
  Then
  $$twp(\alpha, \phi) \subseteq M[\pi \circ \psi'] \subseteq wp(\alpha, \phi)$$
  by lemma 17 1a) and 2a).
- π is odd:
  Analogous by lemma 17 1b) and 2b).

■

Theorem 19 *M is expressive iff for every α the set $wp(\alpha, false)$ is definable.*

Proof: The only if-part is trivial.
To prove the if-part, suppose that for every α the set $wp(\alpha, false)$ is definable. First we show that M is weak expressive. Let b be a formula without quantifiers, and α a program. Define

$$\alpha' = [\ \alpha; \text{ if } b \text{ then while true do } x := x]$$

We have $wp(\alpha', false) = wp(\alpha, b)$. So $wp(\alpha, b)$ is definable. By theorem 18 it follows that M is weak expressive.
But now we have a weak expressive model where weakest preconditions of arbitrary programs and false are definable. So by theorem 10 it follows that M is expressive. ■

Finally we show that complete models are weak expressive.

Theorem 20 *If M is complete, then M is weak expressive.*

Proof: We show that M is weak expressive for formulae b without quantifiers. Then the result follows by theorem 18.
Let α be a program in P, and b a quantifier-free formula. Suppose $x_1, \ldots, x_n$ are the variables occuring in program α, and $y_1, \ldots, y_n$ are *new* variables (i.e. they do not occur in α or b). Define a program as follows:

α'=[$y_1 := x_1; \ldots ; y_n := x_n; \alpha$;if $\neg b$ then while true do $x_1 := x_1$ od fi;$x_1 := y_1; \ldots; x_n := y_n$]

We now have $\models_M \{true\}\alpha'; \alpha\{b\}\}$. By completeness of M it follows that $\vdash_M \{true\}\alpha'; \alpha\{b\}$. Because we can only prove this by the rule of composition we must have $\vdash_M \{true\}\alpha'\{\psi'\}$ and $\vdash_M \{\psi'\}\alpha\{b\}$ for a certain ψ'. So $\models_M \{true\}\alpha'\{\psi'\}$ and $\models_M \{\psi'\}\alpha\{b\}$. By definition of α' we have therefore

$$\sigma \in twp(\alpha, b) \Rightarrow \sigma[\sigma(x_1)/y_1] \ldots [\sigma(x_n)/y_n] \in M[\psi']$$

Hence

$$\sigma \in twp(\alpha, b) \Rightarrow \sigma \in M[\psi'']$$

where ψ'' is the formula obtained by replacing each y_i by x_i in ψ'. So $twp(\alpha, b) \subseteq M[\psi'']$.

By $\models_M \{\psi'\}\alpha\{b\}$ we have that $M[\psi'] \subseteq wp(\alpha, b)$. Suppose $\sigma \in M[\psi'']$, then $\sigma[\sigma(x_1)/y_1] \ldots [\sigma(x_n)/y_n] \in M[\psi']$, so $\sigma[\sigma(x_1)/y_1] \ldots [\sigma(x_n)/y_n] \in wp(\alpha, b)$. We have therefore

$$\forall \sigma'(M[\alpha](\sigma[\sigma(x_1)/y_1] \ldots [\sigma(x_n)/y_n]) = \sigma' \Rightarrow \sigma' \in M[b])$$

By property (1) of the semantics and the fact that α does not contain an y_i, we have

$$\forall \sigma''(M[\alpha](\sigma) = \sigma'' \Rightarrow \sigma''[\sigma(x_1)/y_1] \ldots [\sigma(x_n)/y_n] \in M[b])$$

Because b does not contain an y_i it follows that

$$\forall \sigma''(M[\alpha](\sigma) = \sigma'' \Rightarrow \sigma'' \in M[b])$$

So $\sigma \in wp(\alpha, b)$. So $M[\psi''] \subseteq wp(\alpha, b)$.
We have therefore $twp(\alpha, b) \subseteq M[\psi''] \subseteq wp(\alpha, b)$. ■

Example 21 *Let $\mathcal{L}$ be the language with a unary predicate Z, a unary function s, and a constant c. Take as a model PRED, with as domain the natural numbers, $Z(x)$ iff $x = 0$, $s(x) = x - 1$, $s(0) = 0$, and $c = 0$. Define*

$$\alpha = [\textit{while } \neg Z(x) \wedge \neg Z(s(x)) \textit{ do } x := s(s(x)) \textit{ od}]$$

Observe that α is total, and $wp(\alpha, Z(x)) = \{\sigma | \sigma(x) \textit{is even}\}$. The set of even numbers is not definable in $\mathcal{L}$, hence PRED is not complete by the fact that in a weak expressive model every total program has a definable weakest precondition.

Acknowledgement: I thank Jan van Leeuwen for the critical reading of a draft version of this paper and for giving helpfull comments.

References

[1] Bergstra, J.A., Tucker, J.V.: *Expressiveness and the Completeness of Hoare's Logic*, J. Comput. Syst. Sci. 25, 276-284 (1982)

[2] Loeckx, J., Sieber, K.: *The Foundations of Program Verification*, Wiley-Teubner Series in Computer Science, Stuttgart 1984