Semantics and Verification of Software

Lecture 10: Axiomatic Semantics of WHILE III (Completeness & Total Correctness)

Thomas Noll

Lehrstuhl für Informatik 2
(Software Modeling and Verification)

RWTH AACHEN UNIVERSITY

noll@cs.rwth-aachen.de

http://www-i2.informatik.rwth-aachen.de/i2/svsw13/

Summer Semester 2013

Outline

1. Recapitulation: Hoare Logic
2. (In-)Completeness of Hoare Logic
3. Relative Completeness of Hoare Logic
4. Total Correctness
5. Soundness and Completeness of Total Correctness

Partial Correctness Properties

Validity of property $\{A\}\,c\,\{B\}$

For all states $\sigma \in \Sigma$ which satisfy A:
if the execution of c in σ terminates in $\sigma' \in \Sigma$, then σ' satisfies B.

Hoare Logic

Goal: syntactic derivation of valid partial correctness properties. Here $A[x \mapsto a]$ denotes the syntactic replacement of every occurrence of x by a in A.

Tony Hoare (* 1934)

Definition (Hoare Logic)

The Hoare rules are given by

$$(\text{skip})\,\frac{}{\{A\}\,\texttt{skip}\,\{A\}} \qquad (\text{asgn})\,\frac{}{\{A[x \mapsto a]\}\,x\texttt{:=}a\,\{A\}}$$

$$(\text{seq})\,\frac{\{A\}\,c_1\,\{C\} \quad \{C\}\,c_2\,\{B\}}{\{A\}\,c_1\,\texttt{;}\,c_2\,\{B\}} \qquad (\text{if})\,\frac{\{A \wedge b\}\,c_1\,\{B\} \quad \{A \wedge \neg b\}\,c_2\,\{B\}}{\{A\}\,\texttt{if}\ b\ \texttt{then}\ c_1\ \texttt{else}\ c_2\,\{B\}}$$

$$(\text{while})\,\frac{\{A \wedge b\}\,c\,\{A\}}{\{A\}\,\texttt{while}\ b\ \texttt{do}\ c\,\{A \wedge \neg b\}}$$

$$(\text{cons})\,\frac{\models (A \Rightarrow A') \quad \{A'\}\,c\,\{B'\} \quad \models (B' \Rightarrow B)}{\{A\}\,c\,\{B\}}$$

A partial correctness property is provable (notation: $\vdash \{A\}\,c\,\{B\}$) if it is derivable by the Hoare rules. In (while), A is called a (loop) invariant.

Soundness of Hoare Logic

Soundness: only (semantically) valid partial correctness properties can be (syntactically) derived

Theorem (Soundness of Hoare Logic)

For every partial correctness property $\{A\}\,c\,\{B\}$,

$$\vdash \{A\}\,c\,\{B\} \quad \Rightarrow \quad \models \{A\}\,c\,\{B\}.$$

Proof.

Let $\vdash \{A\}\,c\,\{B\}$. By induction over the structure of the corresponding proof tree we show that, for every $\sigma \in \Sigma$ and $I \in Int$ such that $\sigma \models^I A$, $\mathfrak{C}[\![c]\!]\sigma \models^I B$ (on the board).
(If $\sigma = \bot$, then $\mathfrak{C}[\![c]\!]\sigma = \bot \models^I B$ holds trivially.) □

Outline

Incompleteness of Hoare Logic I

Soundness: only valid partial correctness properties are provable ✓

Completeness: all valid partial correctness properties are systematically derivable ↯

Theorem 10.1 (Gödel's Incompleteness Theorem)

The set of all valid assertions

$$\{A \in Assn \mid \models A\}$$

is not recursively enumerable, i.e., there exists no proof system for Assn in which all valid assertions are systematically derivable.

Kurt Gödel (1906–1978)

Proof.

see [Winskel 1996, p. 110 ff] □

Incompleteness of Hoare Logic II

Corollary 10.2

There is no proof system in which all valid partial correctness properties can be enumerated.

Proof.

Given $A \in Assn$, $\models A$ is obviously equivalent to $\{\mathsf{true}\}\,\texttt{skip}\,\{A\}$. Thus the enumerability of all valid partial correctness properties would imply the enumerability of all valid assertions. □

Remark: alternative proof (using computability theory):
$\{\mathsf{true}\}\, c\, \{\mathsf{false}\}$ is valid iff c does not terminate on any input state. But the set of all non-terminating WHILE statements is not enumerable.

Outline

Relative Completeness of Hoare Logic I

- We will see: actual reason of incompleteness is rule

$$(\text{cons})\ \frac{\models (A \Rightarrow A') \quad \{A'\}\, c\, \{B'\} \quad \models (B' \Rightarrow B)}{\{A\}\, c\, \{B\}}$$

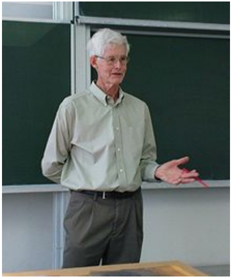
  since it is based on the validity of implications within *Assn*
- The other language constructs are "enumerable"
- Therefore: separation of proof system (Hoare Logic) and assertion language (*Assn*)
- One can show: if an "oracle" is available which decides whether a given assertion is valid, then all valid partial correctness properties can be systematically derived

$\Rightarrow$ Relative completeness

Theorem 10.3 (Cook's Completeness Theorem)

Hoare Logic is relatively complete, i.e., for every partial correctness property $\{A\}\, c\, \{B\}$:

$$\models \{A\}\, c\, \{B\} \quad \Rightarrow \quad \vdash \{A\}\, c\, \{B\}.$$



Stephen A. Cook
(* 1939)

Thus: if we know that a partial correctness property is valid, then we know that there is a corresponding derivation.

The proof uses the following concept: assume that, e.g., $\{A\}\, c_1 ; c_2\, \{B\}$ has to be derived. This requires an intermediate assertion $C \in Assn$ such that $\{A\}\, c_1\, \{C\}$ and $\{C\}\, c_2\, \{B\}$. How to find it?

Weakest Preconditions I

Definition 10.4 (Weakest precondition)

Given $c \in Cmd$, $B \in Assn$ and $I \in Int$, the weakest precondition of B with respect to c under I is defined by:

$$wp^I[\![c, B]\!] := \{\sigma \in \Sigma_\bot \mid \mathfrak{C}[\![c]\!]\sigma \models^I B\}.$$

Corollary 10.5

For every $c \in Cmd$, $A, B \in Assn$, and $I \in Int$:

1. $\models^I \{A\}\, c\, \{B\} \iff A^I \subseteq wp^I[\![c, B]\!]$
2. *If $A_0 \in Assn$ such that $A_0^I = wp^I[\![c, B]\!]$ for every $I \in Int$, then*

$$\models \{A\}\, c\, \{B\} \quad \iff \quad \models (A \Rightarrow A_0)$$

Remark: (2) justifies the notion of weakest precondition: it is implied by every precondition A which makes $\{A\}\, c\, \{B\}$ valid

Weakest Preconditions II

Definition 10.6 (Expressivity of assertion languages)

An assertion language $Assn$ is called expressive if, for every $c \in Cmd$ and $B \in Assn$, there exists $A_{c,B} \in Assn$ such that

$$A^I_{c,B} = wp^I[\![c, B]\!]$$

for every $I \in Int$.

Theorem 10.7 (Expressivity of $Assn$)

$Assn$ is expressive.

Proof.

(idea; see [Winskel 1996, p. 103 ff for details])
Given $c \in Cmd$ and $B \in Assn$, construct $A_{c,B} \in Assn$ with $\sigma \models^I A_{c,B} \iff \mathfrak{C}[\![c]\!]\sigma \models^I B$ (for every $\sigma \in \Sigma_\perp$, $I \in Int$). For example:

$$A_{\texttt{skip},B} := B \qquad A_{x\texttt{:=}a,B} := B[x \mapsto a]$$
$$A_{c_1\texttt{;}c_2,B} := A_{c_1,A_{c_2,B}} \qquad \ldots$$

(for `while`: "Gödelization" of sequences of intermediate states) □

Relative Completeness of Hoare Logic II

The following lemma shows that weakest preconditions are "derivable":

Lemma 10.8

For every $c \in Cmd$ and $B \in Assn$:

$$\vdash \{A_{c,B}\}\, c\, \{B\}$$

Proof.

by structural induction over c (omitted) □

Proof (Cook's Completeness Theorem 10.3).

We have to show that Hoare Logic is relatively complete, i.e., that

$$\models \{A\}\, c\, \{B\} \quad \Rightarrow \quad \vdash \{A\}\, c\, \{B\}.$$

- Lemma 10.8: $\vdash \{A_{c,B}\}\, c\, \{B\}$
- Corollary 10.5: $\models \{A\}\, c\, \{B\} \quad \Rightarrow \quad \models (A \Rightarrow A_{c,B})$
- (cons) $\dfrac{\models (A \Rightarrow A_{c,B}) \quad \{A_{c,B}\}\, c\, \{B\} \quad \models (B \Rightarrow B)}{\{A\}\, c\, \{B\}}$

□

Outline

Total Correctness

- **Observation:** partial correctness properties only speak about terminating computations of a given program
- Total correctness additionally requires the proof that the program indeed stops (on the input states admitted by the precondition)
- Consider total correctness properties of the form

  $$\{A\}\, c\, \{\Downarrow B\}$$

  where $c \in Cmd$ and $A, B \in Assn$
- Interpretation:

Validity of property $\{A\}\, c\, \{\Downarrow B\}$

For all states $\sigma \in \Sigma$ which satisfy A:
the execution of c in σ terminates and yields a state which satisfies B.

Semantics of Total Correctness Properties

Definition 10.9 (Semantics of total correctness properties)

Let $A, B \in Assn$ and $c \in Cmd$.

- $\{A\}\, c\, \{\Downarrow B\}$ is called valid in $\sigma \in \Sigma$ and $I \in Int$ (notation: $\sigma \models^I \{A\}\, c\, \{\Downarrow B\}$) if $\sigma \models^I A$ implies that $\mathfrak{C}[\![c]\!]\sigma \neq \bot$ and $\mathfrak{C}[\![c]\!]\sigma \models^I B$.
- $\{A\}\, c\, \{\Downarrow B\}$ is called valid in $I \in Int$ (notation: $\models^I \{A\}\, c\, \{\Downarrow B\}$) if $\sigma \models^I \{A\}\, c\, \{\Downarrow B\}$ for every $\sigma \in \Sigma$.
- $\{A\}\, c\, \{\Downarrow B\}$ is called valid (notation: $\models \{A\}\, c\, \{\Downarrow B\}$) if $\models^I \{A\}\, c\, \{\Downarrow B\}$ for every $I \in Int$.

Obviously, total implies partial correctness (but not vice versa):

Corollary 10.10

For all $A, B \in Assn$ and $c \in Cmd$,

$$\models \{A\}\, c\, \{\Downarrow B\} \Rightarrow \models \{A\}\, c\, \{B\}.$$

Proving Total Correctness I

Goal: syntactic derivation of valid total correctness properties

Definition 10.11 (Hoare Logic for total correctness)

The Hoare rules for total correctness are given by

$$\text{(skip)}\ \frac{}{\{A\}\ \texttt{skip}\ \{\Downarrow A\}} \qquad \text{(asgn)}\ \frac{}{\{A[x \mapsto a]\}\ x\ \texttt{:=}\ a\ \{\Downarrow A\}}$$

$$\text{(seq)}\ \frac{\{A\}\ c_1\ \{\Downarrow C\} \quad \{C\}\ c_2\ \{\Downarrow B\}}{\{A\}\ c_1 ; c_2\ \{\Downarrow B\}} \qquad \text{(if)}\ \frac{\{A \wedge b\}\ c_1\ \{\Downarrow B\} \quad \{A \wedge \neg b\}\ c_2\ \{\Downarrow B\}}{\{A\}\ \texttt{if}\ b\ \texttt{then}\ c_1\ \texttt{else}\ c_2\ \{\Downarrow B\}}$$

$$\text{(while)}\ \frac{\models (i \geq 0 \wedge A(i+1) \Rightarrow b) \quad \{i \geq 0 \wedge A(i+1)\}\ c\ \{\Downarrow A(i)\} \quad \models (A(0) \Rightarrow \neg b)}{\{\exists i . i \geq 0 \wedge A(i)\}\ \texttt{while}\ b\ \texttt{do}\ c\ \{\Downarrow A(0)\}}$$

$$\text{(cons)}\ \frac{\models (A \Rightarrow A') \quad \{A'\}\ c\ \{\Downarrow B'\} \quad \models (B' \Rightarrow B)}{\{A\}\ c\ \{\Downarrow B\}}$$

where $i \in LVar$.
A total correctness property is provable (notation: $\vdash \{A\}\ c\ \{\Downarrow B\}$) if it is derivable by the Hoare rules. In case of (while), $A(i)$ is called a (loop) invariant.

Proving Total Correctness II

- In rule

$$(\text{while})\ \frac{\models (i \geq 0 \land A(i+1) \Rightarrow b) \quad \{i \geq 0 \land A(i+1)\}\, c\, \{\Downarrow A(i)\} \quad \models (A(0) \Rightarrow \neg b)}{\{\exists i. i \geq 0 \land A(i)\}\, \texttt{while}\ b\ \texttt{do}\ c\, \{\Downarrow A(0)\}}$$

  the notation $A(i)$ indicates that assertion A parametrically depends on the value of the logical variable $i \in LVar$.

- Idea: i represents the remaining number of loop iterations
- Loop to be traversed $i+1$ times ($i \geq 0$)
  $\Rightarrow A(i+1)$ holds
  $\Rightarrow$ execution condition b satisfied

  Thus: $\models (i \geq 0 \land A(i+1) \Rightarrow b)$, and $i+1$ decreased to i after execution of c

- Execution terminated
  $\Rightarrow A(0)$ holds
  $\Rightarrow$ execution condition b violated

  Thus: $\models (A(0) \Rightarrow \neg b)$

Total Correctness of Factorial Program I

Example 10.12

Proof of $\{A\}\,\texttt{y:=1;}c\,\{\Downarrow B\}$ where

$$A := (\mathtt{x} > 0 \wedge \mathtt{x} = i)$$
$$c := \texttt{while } \neg\texttt{(x=1) do (y:=y*x; x:=x-1)}$$
$$B := (\mathtt{y} = i!)$$

First we show that the assertion $C(j) = (\mathtt{x} > 0 \wedge \mathtt{y} * \mathtt{x}! = i! \wedge \mathtt{x} = j+1)$ is an invariant of c. Applying (asgn) twice yields

$\vdash \{j \geq 0 \wedge C(j)[\mathtt{x} \mapsto \mathtt{x-1}]\}\,\texttt{x:=x-1}\,\{\Downarrow j \geq 0 \wedge C(j)\}$ and
$\vdash \{j \geq 0 \wedge C(j)[\mathtt{x} \mapsto \mathtt{x-1}][\mathtt{y} \mapsto \mathtt{y*x}]\}\,\texttt{y:=y*x}\,\{\Downarrow j \geq 0 \wedge C(j)[\mathtt{x} \mapsto \mathtt{x-1}]\}$

such that (seq) implies

$$\vdash \{j \geq 0 \wedge C(j)[\mathtt{x} \mapsto \mathtt{x-1}][\mathtt{y} \mapsto \mathtt{y*x}]\}\,\texttt{y:=y*x; x:=x-1}\,\{\Downarrow j \geq 0 \wedge C(j)\}.$$

Now $C(j+1) = (\mathtt{x} > 0 \wedge \mathtt{y*x}! = i! \wedge \mathtt{x} = j+2)$ and
$C(j)[\mathtt{x} \mapsto \mathtt{x-1}][\mathtt{y} \mapsto \mathtt{y*x}] = (\mathtt{x}-1 > 0 \wedge \mathtt{y} * \mathtt{x} * (\mathtt{x}-1)! = i! \wedge \mathtt{x}-1 = j+1)$
such that

$\models ((j \geq 0 \wedge C(j+1)) \Rightarrow (j \geq 0 \wedge C(j)[\mathtt{x} \mapsto \mathtt{x-1}][\mathtt{y} \mapsto \mathtt{y*x}]))$ and
$\models ((j \geq 0 \wedge C(j)) \Rightarrow C(j)).$

Total Correctness of Factorial Program II

Example 10.12 (continued)

Hence (cons) implies

$$\vdash \{j \geq 0 \wedge C(j+1)\}\, \texttt{y:=y*x;}\ \ \texttt{x:=x-1}\, \{\Downarrow C(j)\}.$$

Moreover we have

$$\models ((j \geq 0 \wedge C(j+1)) \Rightarrow \neg(\mathrm{x} = 1)) \text{ and } \models (C(0) \Rightarrow \neg(\neg(\mathrm{x} = 1)))$$

such that (while) yields

$$\vdash \{\exists j.j \geq 0 \wedge C(j)\}\, c\, \{\Downarrow C(0)\}.$$

For the initializing assignment, (asgn) implies

$$\vdash \{\exists j.j \geq 0 \wedge C(j)[\mathrm{y} \mapsto 1]\}\, \texttt{y:=1}\, \{\Downarrow \exists j.j \geq 0 \wedge C(j)\},$$

such that (seq) allows to conclude

$$\vdash \{\exists j.j \geq 0 \wedge C(j)[\mathrm{y} \mapsto 1]\}\, \texttt{y:=1;}c\, \{\Downarrow C(0)\}.$$

On the other hand we have (choose $j := i - 1$):

$$\models ((\mathrm{x} > 0 \wedge x = i) \Rightarrow (\exists j.j \geq 0 \wedge C(j)[\mathrm{y} \mapsto 1])) \text{ and } \models (C(0) \Rightarrow \mathrm{y} = i!)$$

such that (cons) yields the desired result:

$$\vdash \{\mathrm{x} > 0 \wedge \mathrm{x} = i\}\, \texttt{y:=1;}c\, \{\Downarrow \mathrm{y} = i!\}.$$

Outline

Soundness

In analogy to Theorem 9.4 we can show that the Hoare Logic for total correctness properties is also sound:

Theorem 10.13 (Soundness)

For every total correctness property $\{A\}\,c\,\{\Downarrow B\}$,

$$\vdash \{A\}\,c\,\{\Downarrow B\} \quad \Rightarrow \quad \models \{A\}\,c\,\{\Downarrow B\}.$$

Proof.

again by structural induction over the derivation tree of $\vdash \{A\}\,c\,\{\Downarrow B\}$ (only (while) case; on the board) □

Also the counterpart to Cook's Completeness Theorem 10.3 applies:

Theorem 10.14 (Completeness)

The Hoare Logic for total correctness properties is relatively complete, i.e., for every $\{A\}\, c\, \{\Downarrow B\}$:

$$\models \{A\}\, c\, \{\Downarrow B\} \quad \Rightarrow \quad \vdash \{A\}\, c\, \{\Downarrow B\}.$$

Proof.

omitted □