Semantics and Verification of Software

Lecture 13: Axiomatic Semantics of WHILE IV (Relative Completeness and Total Correctness Properties)

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/svsw10/

Summer Semester 2010

Outline

1. Repetition: Correctness of Hoare Logic
2. Relative Completeness of Hoare Logic
3. Total Correctness

Hoare Logic

Goal: syntactic derivation of valid partial correctness properties

Definition (Hoare Logic)

The Hoare rules are given by

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

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

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

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

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

Here $A[x \mapsto a]$ denotes the syntactic replacement of every occurrence of x by a in A.

Theorem (Soundness of Hoare Logic)

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

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

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.) □

Incompleteness of Hoare Logic II

Corollary

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

1. Repetition: Correctness of Hoare Logic
2. Relative Completeness of Hoare Logic
3. Total Correctness

Relative Completeness of Hoare Logic I

- We will see: actual reason of incompleteness is rule

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

since it is based on the validity of implications within $Assn$

Relative Completeness of Hoare Logic I

- We will see: actual reason of incompleteness is rule

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

  since it is based on the validity of implications within $Assn$
- The other language constructs are "enumerable"

Relative Completeness of Hoare Logic I

- We will see: actual reason of incompleteness is rule

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

  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 ($Assn$)

Relative Completeness of Hoare Logic I

- We will see: actual reason of incompleteness is rule

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

  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

Relative Completeness of Hoare Logic I

- We will see: actual reason of incompleteness is rule

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

  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

$\implies$ Relative completeness

Relative Completeness of Hoare Logic II

Theorem 13.1 (Cook's Completeness Theorem)

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

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

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

Relative Completeness of Hoare Logic II

Theorem 13.1 (Cook's Completeness Theorem)

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

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

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 13.2 (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:

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

Weakest Preconditions I

Definition 13.2 (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:

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

Corollary 13.3

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*
$$\models \{A\}\, c\, \{B\} \iff \models (A \implies A_0)$$

Weakest Preconditions I

Definition 13.2 (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:

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

Corollary 13.3

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

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

Definition 13.4 (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 $I \in Int$.

Definition 13.4 (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 13.5 (Expressivity of $Assn$)

$Assn$ is expressive.

Weakest Preconditions II

Definition 13.4 (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 13.5 (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 13.6

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

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

Relative Completeness of Hoare Logic II

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

Lemma 13.6

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

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

Proof.

by structural induction over c (omitted) □

Relative Completeness of Hoare Logic II

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

Lemma 13.6

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 13.1).

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

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

Relative Completeness of Hoare Logic II

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

Lemma 13.6

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 13.1).

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

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

- Lemma 13.6 $\quad \Longrightarrow \quad \vdash \{A_{c,B}\}\, c\, \{B\}$

Relative Completeness of Hoare Logic II

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

Lemma 13.6

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 13.1).

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

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

- Lemma 13.6 $\Longrightarrow \vdash \{A_{c,B}\}\, c\, \{B\}$
- Corollary 13.3 $\Longrightarrow \models (A \Longrightarrow A_{c,B})$

Relative Completeness of Hoare Logic II

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

Lemma 13.6

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 13.1).

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

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

- Lemma 13.6 $\Longrightarrow$ $\vdash \{A_{c,B}\}\, c\, \{B\}$
- Corollary 13.3 $\Longrightarrow$ $\models (A \Longrightarrow A_{c,B})$
- (cons) rule $\Longrightarrow$ $\vdash \{A\}\, c\, \{B\}$

□

Outline

1 Repetition: Correctness of Hoare Logic

2 Relative Completeness of Hoare Logic

3 Total Correctness

Total Correctness

- **Observation:** partial correctness properties only speak about terminating computations of a given program

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)

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 $A, B \in Assn$

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.

Definition 13.7 (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 $\mathfrak{C}[\![c]\!]\sigma \models^I B$.

Definition 13.7 (Semantics of total correctness properties)

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

- $\{A\}\, c\, \{\Downarrow B\}$ is called valid in $\sigma \in \Sigma$ and $I \in \mathit{Int}$ (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 $I \in \mathit{Int}$ (notation: $\models^I \{A\}\, c\, \{\Downarrow B\}$) if $\sigma \models^I \{A\}\, c\, \{\Downarrow B\}$ for every $\sigma \in \Sigma$.

Semantics of Total Correctness Properties

Definition 13.7 (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: $\models \{A\}\, c\, \{B\}$) if $\models^I \{A\}\, c\, \{\Downarrow B\}$ for every $I \in Int$.

Proving Total Correctness I

Goal: syntactic derivation of valid total correctness properties

Definition 13.8 (Hoare Logic for total correctness)

The Hoare rules for total correctness are given by

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

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

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

$$(\text{cons})\ \frac{\models (A \implies A')\ \ \{A'\}\,c\,\{\Downarrow B'\}\ \models (B' \implies B)}{\{A\}\,c\,\{\Downarrow B\}}$$

where $i \in LVar$, $\models (i \geq 0 \wedge A(i+1) \implies b)$, and $\models (A(0) \implies \neg b)$.
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

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

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

Proving Total Correctness II

- In rule

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

  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

Proving Total Correctness II

- In rule

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

  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
- Execution terminated

  $\implies$ $A(0)$ holds

  $\implies$ execution condition b false

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

Proving Total Correctness II

- In rule

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

  the notation $A(i)$ indicates that assertion A parametrically depends on the value of the logical variable $i \in \mathit{LVar}$.
- Idea: i represents the remaining number of loop iterations
- Execution terminated
  $\implies$ $A(0)$ holds
  $\implies$ execution condition b false

  Thus: $\models (A(0) \implies \neg b)$
- Loop to be traversed $i+1$ times ($i \geq 0$)
  $\implies$ $A(i+1)$ holds
  $\implies$ execution condition b true

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

Total Correctness of Factorial Program

Example 13.9

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

$$\begin{aligned} A &:= (\texttt{x} > 0 \land \texttt{x} = i) \\ c &:= \texttt{while } \neg\texttt{(x=1) do (y:=y*x; x:=x-1)} \\ B &:= (\texttt{y} = i!) \end{aligned}$$

(on the board)