

Semantics and Verification of Software

Lecture 14: Axiomatic Semantics of WHILE V (Total Correctness and Semantic Equivalence)

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

- 1 Repetition: Total Correctness Properties
- 2 Soundness and Completeness of Total Correctness
- 3 Equivalence of Axiomatic and Operational/Denotational Semantics
- 4 Summary: Axiomatic Semantics

Definition (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 $\mathfrak{C}[[c]]\sigma \neq \perp$ 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 $I \in Int$.

Proving Total Correctness

Goal: syntactic derivation of valid total correctness properties

Definition (Hoare Logic for total correctness)

The **Hoare rules** for total correctness are given by

$$\begin{array}{l} \text{(skip)} \frac{}{\{A\} \text{skip} \{\Downarrow A\}} \qquad \text{(asgn)} \frac{}{\{A[x \mapsto a]\} x := a \{\Downarrow A\}} \\ \text{(seq)} \frac{\{A\} c_1 \{\Downarrow C\} \quad \{C\} c_2 \{\Downarrow B\}}{\{A\} c_1 ; c_2 \{\Downarrow B\}} \qquad \text{(if)} \frac{\{A \wedge b\} c_1 \{\Downarrow B\} \quad \{A \wedge \neg b\} c_2 \{\Downarrow B\}}{\{A\} \text{if } b \text{ then } c_1 \text{ else } c_2 \{\Downarrow B\}} \\ \text{(while)} \frac{\{i \geq 0 \wedge A(i+1)\} c \{\Downarrow A(i)\}}{\{\exists i. i \geq 0 \wedge A(i)\} \text{while } b \text{ do } c \{\Downarrow A(0)\}} \\ \text{(cons)} \frac{\models (A \implies A') \quad \{A'\} c \{\Downarrow B'\} \quad \models (B' \implies B)}{\{A\} c \{\Downarrow B\}} \end{array}$$

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**.

- 1 Repetition: Total Correctness Properties
- 2 Soundness and Completeness of Total Correctness
- 3 Equivalence of Axiomatic and Operational/Denotational Semantics
- 4 Summary: Axiomatic Semantics

In analogy to Theorem 12.2 we can show that the Hoare Logic for total correctness properties is also sound:

Theorem 14.1 (Soundness)

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

$$\vdash \{A\} c \{\Downarrow B\} \implies \models \{A\} c \{\Downarrow B\}.$$

In analogy to Theorem 12.2 we can show that the Hoare Logic for total correctness properties is also sound:

Theorem 14.1 (Soundness)

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

$$\vdash \{A\} c \{\Downarrow B\} \implies \models \{A\} c \{\Downarrow B\}.$$

Proof.

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

Relative Completeness

Also the counterpart to Cook's Completeness Theorem 13.1 applies:

Theorem 14.2 (Completeness)

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

$$\models \{A\} c \{\Downarrow B\} \implies \vdash \{A\} c \{\Downarrow B\}.$$

Proof.

omitted



- 1 Repetition: Total Correctness Properties
- 2 Soundness and Completeness of Total Correctness
- 3 Equivalence of Axiomatic and Operational/Denotational Semantics
- 4 Summary: Axiomatic Semantics

Def. 4.1: $\mathfrak{D}[\![\cdot]\!] : Cmd \rightarrow (\Sigma \dashrightarrow \Sigma)$ given by

$$\mathfrak{D}[\![c]\!](\sigma) = \sigma' \iff \langle c, \sigma \rangle \rightarrow \sigma'$$

Def. 4.1: $\mathfrak{D}[\![\cdot]\!]$: $Cmd \rightarrow (\Sigma \dashrightarrow \Sigma)$ given by

$$\mathfrak{D}[\![c]\!](\sigma) = \sigma' \iff \langle c, \sigma \rangle \rightarrow \sigma'$$

Def. 4.2: Two statements $c_1, c_2 \in Cmd$ are called **operationally equivalent** (notation: $c_1 \sim c_2$) if

$$\mathfrak{D}[\![c_1]\!] = \mathfrak{D}[\![c_2]\!].$$

Def. 4.1: $\mathfrak{D}[\![\cdot]\!]$: $Cmd \rightarrow (\Sigma \dashrightarrow \Sigma)$ given by

$$\mathfrak{D}[\![c]\!](\sigma) = \sigma' \iff \langle c, \sigma \rangle \rightarrow \sigma'$$

Def. 4.2: Two statements $c_1, c_2 \in Cmd$ are called **operationally equivalent** (notation: $c_1 \sim c_2$) if

$$\mathfrak{D}[\![c_1]\!] = \mathfrak{D}[\![c_2]\!].$$

Theorem 9.1: For every $c \in Cmd$,

$$\mathfrak{D}[\![c]\!] = \mathfrak{C}[\![c]\!],$$

$$\text{i.e., } \mathfrak{D}[\![\cdot]\!] = \mathfrak{C}[\![\cdot]\!].$$

In the axiomatic semantics, two statements have to be considered equivalent if they are **indistinguishable** w.r.t. partial correctness properties:

Definition 14.3 (Axiomatic equivalence)

Two statements $c_1, c_2 \in Cmd$ are called **axiomatically equivalent** (notation: $c_1 \approx c_2$) if, for all assertions $A, B \in Assn$,

$$\models \{A\} c_1 \{B\} \iff \models \{A\} c_2 \{B\}.$$

Example 14.4

We show that

$\text{while } b \text{ do } c \approx \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip}$

(cf. Lemma 5.1). Let $A, B \in \text{Assn}$:

$\models \{A\} \text{ while } b \text{ do } c \{B\}$

Example 14.4

We show that

$\text{while } b \text{ do } c \approx \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip}$

(cf. Lemma 5.1). Let $A, B \in \text{Assn}$:

$\models \{A\} \text{ while } b \text{ do } c \{B\}$

$\iff \vdash \{A\} \text{ while } b \text{ do } c \{B\} \quad (\text{Theorem 12.2, 13.1})$

Example 14.4

We show that

$\text{while } b \text{ do } c \approx \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip}$

(cf. Lemma 5.1). Let $A, B \in \text{Assn}$:

$\models \{A\} \text{ while } b \text{ do } c \{B\}$

$\iff \vdash \{A\} \text{ while } b \text{ do } c \{B\}$ (Theorem 12.2, 13.1)

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C\} \text{ while } b \text{ do } c \{C \wedge \neg b\}$ (rule (cons))

Example 14.4

We show that

$\text{while } b \text{ do } c \approx \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip}$

(cf. Lemma 5.1). Let $A, B \in \text{Assn}$:

$\models \{A\} \text{ while } b \text{ do } c \{B\}$

$\iff \vdash \{A\} \text{ while } b \text{ do } c \{B\}$ (Theorem 12.2, 13.1)

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C\} \text{ while } b \text{ do } c \{C \wedge \neg b\}$ (rule (cons))

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C \wedge b\} c \{C\}$ (rule (while))

Example 14.4

We show that

`while b do c` $\approx$ `if b then (c;while b do c) else skip`

(cf. Lemma 5.1). Let $A, B \in \text{Assn}$:

$\models \{A\} \text{while } b \text{ do } c \{B\}$

$\iff \vdash \{A\} \text{while } b \text{ do } c \{B\}$ (Theorem 12.2, 13.1)

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C\} \text{while } b \text{ do } c \{C \wedge \neg b\}$ (rule (cons))

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C \wedge b\} c \{C\}$ (rule (while))

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C \wedge b\} c; \text{while } b \text{ do } c \{C \wedge \neg b\}$ (rule (seq)),
 $\vdash \{C \wedge \neg b\} \text{skip} \{C \wedge \neg b\}$ (rule (skip))

Example 14.4

We show that

$\text{while } b \text{ do } c \approx \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip}$

(cf. Lemma 5.1). Let $A, B \in \text{Assn}$:

$\models \{A\} \text{while } b \text{ do } c \{B\}$

$\iff \vdash \{A\} \text{while } b \text{ do } c \{B\}$ (Theorem 12.2, 13.1)

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C\} \text{while } b \text{ do } c \{C \wedge \neg b\}$ (rule (cons))

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C \wedge b\} c \{C\}$ (rule (while))

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C \wedge b\} c; \text{while } b \text{ do } c \{C \wedge \neg b\}$ (rule (seq)),
 $\vdash \{C \wedge \neg b\} \text{skip} \{C \wedge \neg b\}$ (rule (skip))

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C\} \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip} \{C \wedge \neg b\}$ (rule (if))

Example 14.4

We show that

$\text{while } b \text{ do } c \approx \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip}$

(cf. Lemma 5.1). Let $A, B \in \text{Assn}$:

- $\models \{A\} \text{while } b \text{ do } c \{B\}$
- $\iff \vdash \{A\} \text{while } b \text{ do } c \{B\}$ (Theorem 12.2, 13.1)
- $\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C\} \text{while } b \text{ do } c \{C \wedge \neg b\}$ (rule (cons))
- $\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C \wedge b\} c \{C\}$ (rule (while))
- $\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C \wedge b\} c; \text{while } b \text{ do } c \{C \wedge \neg b\}$ (rule (seq)),
 $\vdash \{C \wedge \neg b\} \text{skip} \{C \wedge \neg b\}$ (rule (skip))
- $\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C\} \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip} \{C \wedge \neg b\}$ (rule (if))
- $\iff \vdash \{A\} \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip} \{B\}$ (rule (cons))

Example 14.4

We show that

$\text{while } b \text{ do } c \approx \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip}$

(cf. Lemma 5.1). Let $A, B \in \text{Assn}$:

$\models \{A\} \text{while } b \text{ do } c \{B\}$

$\iff \vdash \{A\} \text{while } b \text{ do } c \{B\}$ (Theorem 12.2, 13.1)

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C\} \text{while } b \text{ do } c \{C \wedge \neg b\}$ (rule (cons))

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C \wedge b\} c \{C\}$ (rule (while))

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C \wedge b\} c; \text{while } b \text{ do } c \{C \wedge \neg b\}$ (rule (seq)),
 $\vdash \{C \wedge \neg b\} \text{skip} \{C \wedge \neg b\}$ (rule (skip))

$\iff \text{ex. } C \in \text{Assn} \text{ such that } \models (A \implies C), \models (C \wedge \neg b \implies B),$
 $\vdash \{C\} \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip} \{C \wedge \neg b\}$ (rule (if))

$\iff \vdash \{A\} \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip} \{B\}$ (rule (cons))

$\iff \models \{A\} \text{if } b \text{ then } (c; \text{while } b \text{ do } c) \text{ else skip} \{B\}$
(Theorem 12.2, 13.1)

Theorem 14.5

Axiomatic and denotational/operational equivalence coincide, i.e., for all $c_1, c_2 \in \text{Cmd}$,

$$c_1 \approx c_2 \iff c_1 \sim c_2.$$

Theorem 14.5

Axiomatic and denotational/operational equivalence coincide, i.e., for all $c_1, c_2 \in \text{Cmd}$,

$$c_1 \approx c_2 \iff c_1 \sim c_2.$$

Proof.

on the board



- 1 Repetition: Total Correctness Properties
- 2 Soundness and Completeness of Total Correctness
- 3 Equivalence of Axiomatic and Operational/Denotational Semantics
- 4 Summary: Axiomatic Semantics

Summary: Axiomatic Semantics

- Formalized by partial/total correctness properties

Summary: Axiomatic Semantics

- Formalized by **partial/total correctness properties**
- Inductively defined by **Hoare Logic** proof rules

Summary: Axiomatic Semantics

- Formalized by **partial/total correctness properties**
- Inductively defined by **Hoare Logic** proof rules
- Technically involved (especially loop invariants)
 $\implies$ machine support (**proof assistants**) indispensable for larger programs

Summary: Axiomatic Semantics

- Formalized by **partial/total correctness properties**
- Inductively defined by **Hoare Logic** proof rules
- Technically involved (especially loop invariants)
 $\implies$ machine support (**proof assistants**) indispensable for larger programs
- **Equivalence** of axiomatic and operational/denotational semantics

Summary: Axiomatic Semantics

- Formalized by **partial/total correctness properties**
- Inductively defined by **Hoare Logic** proof rules
- Technically involved (especially loop invariants)
 $\implies$ machine support (**proof assistants**) indispensable for larger programs
- **Equivalence** of axiomatic and operational/denotational semantics
- **Software engineering** aspect: integrated development of program and proof (cf. assertions in Java)