Semantics and Verification of Software

Lecture 8: Denotational Semantics of WHILE III (Continuous Functions and Fixpoint Theorem)

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: Chain-Complete Partial Orders
2. Monotonic and Continuous Functions
3. The Fixpoint Theorem
4. An Example
5. Summary: Denotational Semantics

Semantics of Statements

Definition (Denotational semantics of statements)

The (denotational) semantic functional for statements,

$$\mathfrak{C}[\![.]\!] : Cmd \to (\Sigma \dashrightarrow \Sigma),$$

is given by:

$$\begin{aligned}
\mathfrak{C}[\![\texttt{skip}]\!] &:= \mathsf{id}_\Sigma \\
\mathfrak{C}[\![x \ \texttt{:=}\ a]\!]\sigma &:= \sigma[x \mapsto \mathfrak{A}[\![a]\!]\sigma] \\
\mathfrak{C}[\![c_1\,\texttt{;}\,c_2]\!] &:= \mathfrak{C}[\![c_2]\!] \circ \mathfrak{C}[\![c_1]\!] \\
\mathfrak{C}[\![\texttt{if}\ b\ \texttt{then}\ c_1\ \texttt{else}\ c_2]\!] &:= \mathsf{cond}(\mathfrak{B}[\![b]\!], \mathfrak{C}[\![c_1]\!], \mathfrak{C}[\![c_2]\!]) \\
\mathfrak{C}[\![\texttt{while}\ b\ \texttt{do}\ c]\!] &:= \mathsf{fix}(\Phi)
\end{aligned}$$

where $\Phi : (\Sigma \dashrightarrow \Sigma) \to (\Sigma \dashrightarrow \Sigma) : f \mapsto \mathsf{cond}(\mathfrak{B}[\![b]\!], f \circ \mathfrak{C}[\![c]\!], \mathsf{id}_\Sigma)$

Characterization of $\mathsf{fix}(\Phi)$ II

Now $\mathsf{fix}(\Phi)$ can be characterized by:

- $\mathsf{fix}(\Phi)$ is a fixpoint of Φ, i.e.,

$$\Phi(\mathsf{fix}(\Phi)) = \mathsf{fix}(\Phi)$$

- $\mathsf{fix}(\Phi)$ is minimal with respect to $\sqsubseteq$, i.e., for every $f_0 : \Sigma \dashrightarrow \Sigma$ such that $\Phi(f_0) = f_0$,

$$\mathsf{fix}(\Phi) \sqsubseteq f_0$$

Example

For `while true do skip` we obtain for every $f : \Sigma \dashrightarrow \Sigma$:

$$\Phi(f) = \mathsf{cond}(\mathfrak{B}[\![\mathsf{true}]\!], f \circ \mathfrak{C}[\![\mathtt{skip}]\!], \mathsf{id}_\Sigma) = f$$

$\Longrightarrow$ $\mathsf{fix}(\Phi) = f_\emptyset$ where $f_\emptyset(\sigma) :=$ undefined for every $\sigma \in \Sigma$ (that is, $\mathsf{graph}(f_\emptyset) = \emptyset$)

Characterization of $\mathsf{fix}(\Phi)$ III

Goals:

- Prove existence of $\mathsf{fix}(\Phi)$ for $\Phi(f) = \mathsf{cond}(\mathfrak{B}[\![b]\!], f \circ \mathfrak{C}[\![c]\!], \mathsf{id}_\Sigma)$
- Show how it can be "computed" (more exactly: approximated)

Sufficient conditions:

on domain $\Sigma \dashrightarrow \Sigma$: chain-complete partial order

on function Φ: continuity

Chains and Least Upper Bounds I

Definition (Chain, (least) upper bound)

Let $(D, \sqsubseteq)$ be a partial order and $S \subseteq D$.

1. S is called a chain in D if, for every $s_1, s_2 \in S$,
$$s_1 \sqsubseteq s_2 \text{ or } s_2 \sqsubseteq s_1$$
(that is, S is a totally ordered subset of D).
2. An element $d \in D$ is called an upper bound of S if $s \sqsubseteq d$ for every $s \in S$ (notation: $S \sqsubseteq d$).
3. An upper bound d of S is called least upper bound (LUB) or supremum of S if $d \sqsubseteq d'$ for every upper bound d' of S (notation: $d = \sqcup S$).

Chains and Least Upper Bounds II

Example

1. Every subset $S \subseteq \mathbb{N}$ is a chain in $(\mathbb{N}, \leq)$.
   It has a LUB (its greatest element) iff it is finite.
2. $\{\emptyset, \{0\}, \{0,1\}, \ldots\}$ is a chain in $(2^{\mathbb{N}}, \subseteq)$ with LUB $\mathbb{N}$.
3. Let $x \in Var$, and let $f_i : \Sigma \dashrightarrow \Sigma$ for every $i \in \mathbb{N}$ be given by

$$f_i(\sigma) := \begin{cases} \sigma[x \mapsto \sigma(x)+1] & \text{if } \sigma(x) \leq i \\ \text{undefined} & \text{otherwise} \end{cases}$$

   Then $\{f_0, f_1, f_2, \ldots\}$ is a chain in $(\Sigma \dashrightarrow \Sigma, \sqsubseteq)$, since for every $i \in \mathbb{N}$ and $\sigma, \sigma' \in \Sigma$:

$$\begin{aligned} & f_i(\sigma) = \sigma' \\ \Longrightarrow\quad & \sigma(x) \leq i, \sigma' = \sigma[x \mapsto \sigma(x)+1] \\ \Longrightarrow\quad & \sigma(x) \leq i+1, \sigma' = \sigma[x \mapsto \sigma(x)+1] \\ \Longrightarrow\quad & f_{i+1}(\sigma) = \sigma' \\ \Longrightarrow\quad & f_i \sqsubseteq f_{i+1} \end{aligned}$$

Chain Completeness

Definition (Chain completeness)

A partial order is called chain complete (CCPO) if every of its chains has a least upper bound.

Example

1. $(2^{\mathbb{N}}, \subseteq)$ is a CCPO with $\sqcup S = \bigcup_{M \in S} M$ for every chain $S \subseteq 2^{\mathbb{N}}$.
2. $(\mathbb{N}, \leq)$ is not chain complete (since, e.g., the chain $\mathbb{N}$ has no upper bound).

Application to fix(Φ)

Lemma

- $(\Sigma \dashrightarrow \Sigma, \sqsubseteq)$ *is a CCPO with least element* $f_\emptyset$ *where* $\mathsf{graph}(f_\emptyset) = \emptyset$.
- *In particular, for every chain* $S \subseteq \Sigma \dashrightarrow \Sigma$,

$$\mathsf{graph}\,(\sqcup S) = \bigcup_{f \in S} \mathsf{graph}(f).$$

Proof.

on the board □

Example

Let $x \in Var$, and let $f_i : \Sigma \dashrightarrow \Sigma$ for every $i \in \mathbb{N}$ be given by

$$f_i(\sigma) := \begin{cases} \sigma[x \mapsto \sigma(x) + 1] & \text{if } \sigma(x) \leq i \\ \text{undefined} & \text{otherwise} \end{cases}$$

Then $\sqcup\{f_0, f_1, f_2, \ldots\} = f$ where

$$f : \Sigma \to \Sigma : \sigma \mapsto \sigma[x \mapsto \sigma(x) + 1]$$

Outline

Monotonicity I

Definition 8.1 (Monotonicity)

Let $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$ be partial orders, and let $F : D \to D'$. F is called monotonic (w.r.t. $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$) if, for every $d_1, d_2 \in D$,

$$d_1 \sqsubseteq d_2 \implies F(d_1) \sqsubseteq' F(d_2).$$

Interpretation: monotonic functions "preserve information"

Example 8.2

1. Let $T := \{S \subseteq \mathbb{N} \mid S \text{ finite}\}$. Then $F_1 : T \to \mathbb{N} : S \mapsto \sum_{n \in S} n$ is monotonic w.r.t. $(2^{\mathbb{N}}, \subseteq)$ and $(\mathbb{N}, \leq)$.
2. $F_2 : 2^{\mathbb{N}} \to 2^{\mathbb{N}} : S \mapsto \mathbb{N} \setminus S$ is not monotonic w.r.t. $(2^{\mathbb{N}}, \subseteq)$ (since, e.g., $\emptyset \subseteq \mathbb{N}$ but $F_2(\emptyset) = \mathbb{N} \not\subseteq F_2(\mathbb{N}) = \emptyset$).

Application to fix(Φ)

Lemma 8.3

Let $b \in BExp$, $c \in Cmd$, and $\Phi : (\Sigma \dashrightarrow \Sigma) \to (\Sigma \dashrightarrow \Sigma)$ with $\Phi(f) := \mathsf{cond}(\mathfrak{B}[\![b]\!], f \circ \mathfrak{C}[\![c]\!], \mathsf{id}_\Sigma)$. Then Φ is monotonic w.r.t. $(\Sigma \dashrightarrow \Sigma, \sqsubseteq)$.

Proof.

on the board □

Monotonicity II

The following lemma states how chains behave under monotonic functions.

Lemma 8.4

Let $(D, \sqsubseteq)$ *and* $(D', \sqsubseteq')$ *be CCPOs,* $F : D \to D'$ *monotonic, and* $S \subseteq D$ *a chain in* D*. Then:*

1. $F(S) := \{F(d) \mid d \in S\}$ *is a chain in* D'*.*
2. $\sqcup F(S) \sqsubseteq' F(\sqcup S)$*.*

Proof.

on the board □

Continuity

A function F is continuous if applying F and taking LUBs can be exchanged:

Definition 8.5 (Continuity)

Let $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$ be CCPOs and $F : D \to D'$ monotonic. Then F is called continuous (w.r.t. $(D, \sqsubseteq)$ and $(D', \sqsubseteq')$) if, for every non-empty chain $S \subseteq D$,

$$F\left(\sqcup S\right) = \sqcup F(S).$$

Lemma 8.6

Let $b \in BExp$, $c \in Cmd$, *and* $\Phi(f) := \mathsf{cond}(\mathfrak{B}[\![b]\!], f \circ \mathfrak{C}[\![c]\!], \mathsf{id}_\Sigma)$. *Then* Φ *is continuous w.r.t.* $(\Sigma \dashrightarrow \Sigma, \sqsubseteq)$.

Proof.

omitted □

Outline

Theorem 8.7 (Fixpoint Theorem by Tarski and Knaster)

Let $(D, \sqsubseteq)$ be a CCPO and $F : D \to D$ continuous. Then

$$\mathsf{fix}(F) := \bigsqcup \{F^n(\bigsqcup\emptyset) \mid n \in \mathbb{N}\}$$

is the least fixpoint of F where

$$F^0(d) := d \text{ and } F^{n+1}(d) := F(F^n(d)).$$

Proof.

on the board (later) □

Application to fix(Φ)

Altogether this completes the definition of $\mathfrak{C}[\![.]\!]$. In particular, for the `while` statement we obtain:

Corollary 8.8

Let $b \in \mathit{BExp}$, $c \in \mathit{Cmd}$, *and* $\Phi(f) := \mathsf{cond}(\mathfrak{B}[\![b]\!], f \circ \mathfrak{C}[\![c]\!], \mathsf{id}_\Sigma)$. *Then*

$$\mathsf{graph}(\mathsf{fix}(\Phi)) = \bigcup_{n \in \mathbb{N}} \mathsf{graph}(\Phi^n(f_\emptyset))$$

Proof.

Using

- Lemma 7.9
  - $(\Sigma \dashrightarrow \Sigma, \sqsubseteq)$ CCPO with least element $f_\emptyset$
  - LUB = union of graphs
- Lemma 8.6 (Φ continuous)
- Theorem 8.7 (Fixpoint Theorem)

□

Outline

Example 8.9 (Factorial program)

- Let $c \in Cmd$ be given by

  $$\texttt{y:=1; while } \neg\texttt{(x=1) do (y:=y*x; x:=x-1)}$$

- For every initial state $\sigma_0 \in \Sigma$, Def. 5.4 yields:

  $$\mathfrak{C}[\![c]\!](\sigma_0) = \mathsf{fix}(\Phi)(\sigma_1)$$

  where $\sigma_1 := \sigma_0[\mathtt{y} \mapsto 1]$ and, for every $f : \Sigma \dashrightarrow \Sigma$ and $\sigma \in \Sigma$,

  $$\begin{aligned}\Phi(f)(\sigma) &= \mathsf{cond}(\mathfrak{B}[\![\neg\texttt{(x=1)}]\!], f \circ \mathfrak{C}[\![\texttt{y:=y*x; x:=x-1}]\!], \mathsf{id}_\Sigma)(\sigma) \\ &= \begin{cases}\sigma & \text{if } \sigma(\mathtt{x}) = 1 \\ f(\sigma') & \text{otherwise}\end{cases}\end{aligned}$$

  with $\sigma' := \sigma[\mathtt{y} \mapsto \sigma(\mathtt{y}) * \sigma(\mathtt{x}), \mathtt{x} \mapsto \sigma(\mathtt{x}) - 1]$.

- Approximations of least fixpoint of Φ according to Theorem 8.7:

  $$\mathsf{fix}(\Phi) = \bigsqcup\{\Phi^n(f_\emptyset) \mid n \in \mathbb{N}\}$$

  (where $\mathsf{graph}(f_\emptyset) = \emptyset$)

Example 8.9 (Factorial program; continued)

$$
\begin{aligned}
f_0(\sigma) &:= \Phi^0(f_\emptyset)(\sigma)\\
&= f_\emptyset(\sigma)\\
&= \text{undefined}
\end{aligned}
$$

$$
\begin{aligned}
f_1(\sigma) &:= \Phi^1(f_\emptyset)(\sigma)\\
&= \Phi(f_0)(\sigma)\\
&= \begin{cases}\sigma & \text{if } \sigma(\mathtt{x}) = 1\\ f_0(\sigma') & \text{otherwise}\end{cases}\\
&= \begin{cases}\sigma & \text{if } \sigma(\mathtt{x}) = 1\\ \text{undefined} & \text{otherwise}\end{cases}
\end{aligned}
$$

$$
\begin{aligned}
f_2(\sigma) &:= \Phi^2(f_\emptyset)(\sigma)\\
&= \Phi(f_1)(\sigma)\\
&= \begin{cases}\sigma & \text{if } \sigma(\mathtt{x}) = 1\\ f_1(\sigma') & \text{otherwise}\end{cases}\\
&= \begin{cases}\sigma & \text{if } \sigma(\mathtt{x}) = 1\\ \sigma' & \text{if } \sigma(\mathtt{x}) \neq 1 \text{ and } \sigma'(\mathtt{x}) = 1\\ \text{undefined} & \text{if } \sigma(\mathtt{x}) \neq 1 \text{ and } \sigma'(\mathtt{x}) \neq 1\end{cases}\\
&= \begin{cases}\sigma & \text{if } \sigma(\mathtt{x}) = 1\\ \sigma' & \text{if } \sigma(\mathtt{x}) = 2\\ \text{undefined} & \text{if } \sigma(\mathtt{x}) \neq 1 \text{ and } \sigma(\mathtt{x}) \neq 2\end{cases}\\
&= \begin{cases}\sigma & \text{if } \sigma(\mathtt{x}) = 1\\ \sigma[y \mapsto 2 * \sigma(\mathtt{y}), & \text{if } \sigma(\mathtt{x}) = 2\\ \quad \mathtt{x} \mapsto 1] & \\ \text{undefined} & \text{if } \sigma(\mathtt{x}) \neq 1\\ & \text{and } \sigma(\mathtt{x}) \neq 2\end{cases}
\end{aligned}
$$

Example 8.9 (Factorial program; continued)

$$
\begin{aligned}
f_3(\sigma) &:= \Phi^3(f_\emptyset)(\sigma) \\
&= \Phi(f_2)(\sigma) \\
&= \begin{cases} \sigma & \text{if } \sigma(\mathrm{x}) = 1 \\ f_2(\sigma') & \text{otherwise} \end{cases} \\
&= \begin{cases} \sigma & \text{if } \sigma(\mathrm{x}) = 1 \\ \sigma' & \text{if } \sigma(\mathrm{x}) \neq 1 \text{ and } \sigma'(\mathrm{x}) = 1 \\ \sigma'[\mathrm{y} \mapsto 2 * \sigma'(\mathrm{y}), \mathrm{x} \mapsto 1] & \text{if } \sigma(\mathrm{x}) \neq 1 \text{ and } \sigma'(\mathrm{x}) = 2 \\ \text{undefined} & \text{if } \sigma(\mathrm{x}) \neq 1 \text{ and } \sigma'(\mathrm{x}) \neq 1 \text{ and } \sigma'(\mathrm{x}) \neq 2 \end{cases} \\
&= \begin{cases} \sigma & \text{if } \sigma(\mathrm{x}) = 1 \\ \sigma' & \text{if } \sigma(\mathrm{x}) = 2 \\ \sigma'[\mathrm{y} \mapsto 2 * \sigma'(\mathrm{y}), \mathrm{x} \mapsto 1] & \text{if } \sigma(\mathrm{x}) = 3 \\ \text{undefined} & \text{if } \sigma(\mathrm{x}) \notin \{1, 2, 3\} \end{cases} \\
&= \begin{cases} \sigma & \text{if } \sigma(\mathrm{x}) = 1 \\ \sigma[\mathrm{y} \mapsto 2 * \sigma(\mathrm{y}), \mathrm{x} \mapsto 1] & \text{if } \sigma(\mathrm{x}) = 2 \\ \sigma[\mathrm{y} \mapsto 3 * 2 * \sigma(\mathrm{y}), \mathrm{x} \mapsto 1] & \text{if } \sigma(\mathrm{x}) = 3 \\ \text{undefined} & \text{if } \sigma(\mathrm{x}) \notin \{1, 2, 3\} \end{cases}
\end{aligned}
$$

Example 8.9 (Factorial program; continued)

- n-th approximation:

$$\begin{aligned} & f_n(\sigma) \\ & := \Phi^n(f_\emptyset)(\sigma) \\ & = \begin{cases} \sigma[\mathtt{y} \mapsto \sigma(\mathtt{x}) * (\sigma(\mathtt{x}) - 1) * \ldots * 2 * \sigma(\mathtt{y}), & \text{if } 1 \leq \sigma(\mathtt{x}) \leq n \\ \quad \mathtt{x} \mapsto 1] & \\ \text{undefined} & \text{if } \sigma(\mathtt{x}) \notin \{1, \ldots, n\} \end{cases} \\ & = \begin{cases} \sigma[\mathtt{y} \mapsto (\sigma(\mathtt{x}))! * \sigma(\mathtt{y}), \mathtt{x} \mapsto 1] & \text{if } 1 \leq \sigma(\mathtt{x}) \leq n \\ \text{undefined} & \text{if } \sigma(\mathtt{x}) \notin \{1, \ldots, n\} \end{cases} \end{aligned}$$

- Fixpoint:

$$\mathfrak{C}[\![c]\!](\sigma_0) = \mathsf{fix}(\Phi)(\sigma_1) = \begin{cases} \sigma[\mathtt{y} \mapsto (\sigma(\mathtt{x}))!, \mathtt{x} \mapsto 1] & \text{if } \sigma(\mathtt{x}) \geq 1 \\ \text{undefined} & \text{otherwise} \end{cases}$$

Outline

Summary: Denotational Semantics

- Semantic model: partial state transformations $(\Sigma \dashrightarrow \Sigma)$
- Compositional definition of functional $\mathfrak{C}[\![.]\!] : \mathit{Cmd} \to (\Sigma \dashrightarrow \Sigma)$
- Capturing the recursive nature of loops by a fixpoint definition (for a continuous function on a CCPO)
- Approximation by fixpoint iteration