Korrekte Software: Grundlagen und Methoden

Vorlesung 4 vom 23.04.19

Äquivalenz der Operationalen und Denotationalen Semantik

Serge Autexier, Christoph Lüth

Universität Bremen

Sommersemester 2019

Fahrplan

- Einführung
- Operationale Semantik
- Denotationale Semantik
- Äquivalenz der Operationalen und Denotationalen Semantik
- Der Floyd-Hoare-Kalkül
- Invarianten und die Korrektheit des Floyd-Hoare-Kalküls
- Strukturierte Datentypen
- Verifikationsbedingungen
- Vorwärts mit Floyd und Hoare
- Modellierung
- Spezifikation von Funktionen
- Referenzen und Speichermodelle
- Funktionsaufrufe und das Framing-Problem
- Ausblick und Rückblick

Operationale vs. denotationale Semantik

	Operational $\langle a, \sigma \rangle \to_{Aexp} n$	**Denotational** $\mathcal{A}[\![a]\!]$
m$\in$ **Z**	$\langle m, \sigma \rangle \to_{Aexp} m$	$\{(\sigma, m) \mid \sigma \in \Sigma\}$
x$\in$ **Loc**	$\dfrac{x \in Dom(\sigma)}{\langle x, \sigma \rangle \to_{Aexp} \sigma(x)}$	$\{(\sigma, \sigma(x)) \mid \sigma \in \Sigma, x \in Dom(\sigma)\}$
	$\dfrac{x \notin Dom(\sigma)}{\langle x, \sigma \rangle \to_{Aexp} \bot}$	
$a_1 \circ a_2$	$\dfrac{\langle a_1, \sigma \rangle \to_{Aexp} n \quad \langle a_2, \sigma \rangle \to_{Aexp} m \quad n, m \neq \bot}{\langle a_1 \circ a_2, \sigma \rangle \to_{Aexp} n \circ' m}$	$\{(\sigma, n \circ' m) \mid \sigma \in \Sigma, (\sigma, n) \in \mathcal{A}[\![a_1]\!], (\sigma, m) \in \mathcal{A}[\![a_2]\!]\}$
	$\dfrac{\langle a_1, \sigma \rangle \to_{Aexp} n \quad \langle a_2, \sigma \rangle \to_{Aexp} m \quad n = \bot \text{ oder } m = \bot}{\langle a_1 \circ a_2, \sigma \rangle \to_{Aexp} \bot}$	
	$\circ \in \{+, *, -\}$	

Operationale vs. denotationale Semantik

Operational $\langle a, \sigma \rangle \to_{Aexp} n$

$$\frac{\langle a_1, \sigma \rangle \to_{Aexp} n \quad \langle a_2, \sigma \rangle \to_{Aexp} m \quad m \neq 0 \quad m, n \neq \bot}{\langle a_1 \circ a_2, \sigma \rangle \to_{Aexp} n \circ' m}$$

a_1 / a_2

$$\frac{\langle a_1, \sigma \rangle \to_{Aexp} n \quad \langle a_2, \sigma \rangle \to_{Aexp} m \quad n = \bot, m = \bot \text{ oder } m = 0}{\langle a_1 / a_2, \sigma \rangle \to_{Aexp} \bot}$$

Denotational $\mathcal{A}[\![a]\!]$

$$\{(\sigma, n/m) | \sigma \in \Sigma, (\sigma, n) \in \mathcal{A}[\![a_1]\!], (\sigma, m) \in \mathcal{A}[\![a_2]\!], m \neq 0\}$$

Äquivalenz operationale und denotationale Semantik

- Für alle $a \in \mathbf{Aexp}$, für alle $n \in \mathbb{Z}$, für alle Zustände σ:

$$\langle a, \sigma \rangle \rightarrow_{Aexp} n \Leftrightarrow (\sigma, n) \in \mathcal{A}[\![a]\!]$$

$$\langle a, \sigma \rangle \rightarrow_{Aexp} \perp \Leftrightarrow \sigma \notin Dom(\mathcal{A}[\![a]\!])$$

- Beweis Prinzip?

Äquivalenz operationale und denotationale Semantik

- Für alle $a \in \mathbf{Aexp}$, für alle $n \in \mathbb{Z}$, für alle Zustände σ:

$$\langle a, \sigma \rangle \rightarrow_{Aexp} n \Leftrightarrow (\sigma, n) \in \mathcal{A}[\![a]\!]$$

$$\langle a, \sigma \rangle \rightarrow_{Aexp} \perp \Leftrightarrow \sigma \notin Dom(\mathcal{A}[\![a]\!])$$

- Beweis per struktureller Induktion über a. (Warum?)

Operationale vs. denotationale Semantik

	Operational $\langle b, \sigma \rangle \to_{Bexp} \mathbf{0} \mid \mathbf{1}$	Denotational $\mathcal{B}[\![b]\!]$
$\mathbf{1}$	$\langle \mathbf{1}, \sigma \rangle \to_{Bexp} \mathbf{1}$	$\{(\sigma, \mathbf{1}) \mid \sigma \in \Sigma\}$
$\mathbf{0}$	$\langle \mathbf{0}, \sigma \rangle \to_{Bexp} \mathbf{0}$	$\{(\sigma, \mathbf{0}) \mid \sigma \in \Sigma\}$

Operationale vs. denotationale Semantik

	Operat. $\langle b, \sigma \rangle \to_{Bexp} t$	**Denotational** $\mathcal{B}[\![b]\!]$
$a_0 == a_1$	$\dfrac{\langle a_0, \sigma \rangle \to_{Aexp} n \quad \langle a_1, \sigma \rangle \to_{Aexp} m \quad n, m \neq \bot \quad n = m}{\langle a_0 == a_1, \sigma \rangle \to_{Bexp} \mathbf{1}}$ $\dfrac{\langle a_0, \sigma \rangle \to_{Aexp} n \quad \langle a_1, \sigma \rangle \to_{Aexp} m \quad n, m \neq \bot \quad n \neq m}{\langle a_0 == a_1, \sigma \rangle \to_{Bexp} \mathbf{0}}$ $\dfrac{\langle a_0, \sigma \rangle \to_{Aexp} n \quad \langle a_1, \sigma \rangle \to_{Aexp} m \quad n = \bot \text{ oder } m = \bot}{\langle a_0 == a_1, \sigma \rangle \to_{Bexp} \bot}$	$\{(\sigma, \mathbf{1}) \mid \sigma \in \Sigma, (\sigma, n_0) \in \mathcal{A}[\![a_0]\!], (\sigma, n_1) \in \mathcal{A}[\![a_1]\!], n_0 = n_1 \}$ $\cup$ $\{(\sigma, \mathbf{0}) \mid \sigma \in \Sigma, (\sigma, n_0) \in \mathcal{A}[\![a_0]\!], (\sigma, n_1) \in \mathcal{A}[\![a_1]\!], n_0 \neq n_1 \}$
$a1 <= a2$	analog	

Operationale vs. denotationale Semantik

	Operational $\langle a, \sigma \rangle \to_{Bexp} b$	**Denotational** $\mathcal{B}[\![b]\!]$
$b_1 \&\& b_0$	$\dfrac{\langle b_1, \sigma \rangle \to_{Bexp} \mathbf{0}}{\langle b_1 \&\& b_2, \sigma \rangle \to \mathbf{0}}$	$\{(\sigma, \mathbf{0}) \mid (\sigma, \mathbf{0}) \in \mathcal{B}[\![b_1]\!]\}$
	$\dfrac{\langle b_1, \sigma \rangle \to_{Bexp} \mathbf{1} \quad \langle b_2, \sigma \rangle \to_{Bexp} b}{\langle b_1 \&\& b_2, \sigma \rangle \to b}$	$\{(\sigma, b) \mid (\sigma, \mathbf{1}) \in \mathcal{B}[\![b_1]\!], (\sigma, b) \in \mathcal{B}[\![b_2]\!]\}$
	$\dfrac{\langle b_1, \sigma \rangle \to_{Bexp} \bot}{\langle b_1 \&\& b_2, \sigma \rangle \to \bot}$	
$b_1 \mid\mid b_2$		analog
$!n$	...	

Äquivalenz operationale und denotationale Semantik

- Für alle $b \in \mathbf{Bexp}$, für alle $t \in \mathbb{B}$, for alle Zustände σ:

$$\langle b, \sigma \rangle \rightarrow_{Bexp} t \Leftrightarrow (\sigma, t) \in \mathcal{B}[\![b]\!]$$

$$\langle b, \sigma \rangle \rightarrow_{Bexp} \perp \Leftrightarrow \sigma \notin Dom(\mathcal{B}[\![b]\!])$$

- Beweis Prinzip?

Äquivalenz operationale und denotationale Semantik

- Für alle $b \in \mathbf{Bexp}$, für alle $t \in \mathbb{B}$, for alle Zustände σ:

$$\langle b, \sigma \rangle \to_{Bexp} t \Leftrightarrow (\sigma, t) \in \mathcal{B}[\![b]\!]$$

$$\langle b, \sigma \rangle \to_{Bexp} \bot \Leftrightarrow \sigma \notin Dom(\mathcal{B}[\![b]\!])$$

- Beweis per struktureller Induktion über b (unter Verwendung der Äquivalenz für AExp). (Warum?)

Operationale vs. denotationale Semantik

	Operational	Denotational $\mathcal{C}[\![c]\!]$
$\{\}$	$\dfrac{\langle c,\sigma\rangle \to_{Stmt} \sigma' \mid \bot}{\langle \{\},\sigma\rangle \to_{Stmt} \sigma}$	$\mathcal{C}[\![\{\}]\!] = Id$
$c_1; c_2$	$\dfrac{\langle c_1,\sigma\rangle \to_{Stmt} \sigma' \neq \bot \quad \langle c_2,\sigma'\rangle \to_{Stmt} \sigma''}{\langle c_1; c_2,\sigma\rangle \to_{Stmt} \sigma''}$ $\dfrac{\langle c_1,\sigma\rangle \to_{Stmt} \bot}{\langle c_1; c_2,\sigma\rangle \to_{Stmt} \bot}$	$\mathcal{C}[\![c_2]\!] \circ \mathcal{C}[\![c_1]\!]$
$x = a$	$\dfrac{\langle a,\sigma\rangle \to_{Aexp} n}{\langle x = a,\sigma\rangle \to_{Stmt} \sigma[n/x]}$ $\dfrac{\langle a,\sigma\rangle \to_{Aexp} \bot}{\langle x = a,\sigma\rangle \to_{Stmt} \bot}$	$\{(\sigma,\sigma[n/x]) \mid (\sigma,n) \in \mathcal{A}[\![a]\!]\}$

Operationale vs. denotationale Semantik

	Operational	Denotational $\mathcal{C}[\![c]\!]$
	$\langle c, \sigma \rangle \to_{Stmt} \sigma' \mid \bot$	
	$\dfrac{\langle b, \sigma \rangle \to_{Bexp} \bot}{\langle c, \sigma \rangle \to_{Stmt} \bot}$	
if (b) c_0	$\dfrac{\langle b, \sigma \rangle \to_{Bexp} \mathbf{1} \quad \langle c_0, \sigma \rangle \to_{Stmt} \sigma'}{\langle c, \sigma \rangle \to_{Stmt} \sigma'}$	$\{(\sigma, \sigma') \mid (\sigma, \mathbf{1}) \in \mathcal{B}[\![b]\!], (\sigma, \sigma') \in \mathcal{C}[\![c_0]\!]\}$
else c_1	$\dfrac{\langle b, \sigma \rangle \to_{Bexp} \mathbf{0} \quad \langle c_1, \sigma \rangle \to_{Stmt} \sigma'}{\langle c, \sigma \rangle \to_{Stmt} \sigma'}$	$\{(\sigma, \sigma') \mid (\sigma, \mathbf{0}) \in \mathcal{B}[\![b]\!], (\sigma, \sigma') \in \mathcal{C}[\![c_1]\!]\}$

Operationale vs. denotationale Semantik

Operational $\langle c, \sigma \rangle \to_{Stmt} \sigma' | \bot$ **Denotational** $\mathcal{C}[\![c]\!]$

$\underbrace{\textbf{while}\ (b)\ c}_{w}$

$$\frac{\langle b, \sigma \rangle \to_{Bexp} \mathbf{0}}{\langle w, \sigma \rangle \to_{Stmt} \sigma} \qquad \frac{\langle b, \sigma \rangle \to_{Bexp} \bot}{\langle w, \sigma \rangle \to_{Stmt} \bot}$$

$fix(\Gamma)$

$$\frac{\langle b, \sigma \rangle \to_{Bexp} \mathbf{1} \quad \langle c, \sigma \rangle \to_{Stmt} \sigma' \neq \bot \quad \langle w, \sigma' \rangle \to_{Stmt} \sigma''}{\langle w, \sigma \rangle \to_{Stmt} \sigma''}$$

$$\frac{\langle b, \sigma \rangle \to_{Bexp} \mathbf{1} \quad \langle c, \sigma \rangle \to_{Stmt} \bot}{\langle w, \sigma \rangle \to_{Stmt} \bot}$$

mit

$$\begin{aligned} \Gamma(\varphi) \quad = \quad & \{(\sigma, \sigma') \mid (\sigma, \mathbf{1}) \in \mathcal{B}[\![b]\!], (\sigma, \sigma') \in \varphi \circ \mathcal{C}[\![c]\!]\} \\ & \cup \{(\sigma, \sigma) \mid (\sigma, \mathbf{0}) \in \mathcal{B}[\![b]\!]\} \end{aligned}$$

Äquivalenz operationale und denotationale Semantik

- Für alle $c \in \mathbf{Stmt}$, für alle Zustände σ, σ':

$$\langle c, \sigma \rangle \to_{Stmt} \sigma' \Leftrightarrow (\sigma, \sigma') \in \mathcal{C}[\![c]\!]$$

$$\langle c, \sigma \rangle \to_{Stmt} \bot \Rightarrow \sigma \notin Dom(\mathcal{C}[\![c]\!])$$

- $\Rightarrow$ Beweis Prinzip?
- $\Leftarrow$ Beweis Prinzip?

Äquivalenz operationale und denotationale Semantik

- Für alle $c \in \mathbf{Stmt}$, für alle Zustände σ, σ':

$$\langle c, \sigma \rangle \to_{Stmt} \sigma' \Leftrightarrow (\sigma, \sigma') \in \mathcal{C}[\![c]\!]$$

$$\langle c, \sigma \rangle \to_{Stmt} \bot \Rightarrow \sigma \notin Dom(\mathcal{C}[\![c]\!])$$

- $\Rightarrow$ Beweis per Induktion über die Ableitung in der operationalen Semantik (Warum?)

- $\Leftarrow$ Beweis Prinzip?

Äquivalenz operationale und denotationale Semantik

- Für alle $c \in \mathbf{Stmt}$, für alle Zustände σ, σ':

$$\langle c, \sigma \rangle \to_{Stmt} \sigma' \Leftrightarrow (\sigma, \sigma') \in \mathcal{C}[\![c]\!]$$

$$\langle c, \sigma \rangle \to_{Stmt} \bot \Rightarrow \sigma \notin Dom(\mathcal{C}[\![c]\!])$$

- $\Rightarrow$ Beweis per Induktion über die Ableitung in der operationalen Semantik (Warum?)

- $\Leftarrow$ Beweis per struktureller Induktion über c (Verwendung der Äquivalenz für arithmetische und boolsche Ausdrücke). Für die While-Schleife Rückgriff auf Definition des Fixpunkts und Induktion über die Teilmengen $\Gamma^i(\emptyset)$ des Fixpunkts. (Warum?)

Knackpunkt

$$\mathcal{C}[\![w]\!] = \mathit{fix}(\Gamma) = \Gamma(\mathit{fix}(\Gamma)) = \Gamma(\bigcup_{i\geq 0} \Gamma^i(\emptyset)) = \bigcup_{i\geq 0} \Gamma(\Gamma^i(\emptyset))$$

$$= \bigcup_{i\geq 0} \{(\sigma,\sigma') \mid (\sigma,\mathbf{1}) \in \mathcal{B}[\![b]\!], (\sigma,\sigma'') \in \mathcal{C}[\![c]\!], (\sigma'',\sigma') \in \Gamma^i(\emptyset)\}$$

$$\cup \{(\sigma,\sigma) \mid (\sigma,\mathbf{0}) \in \mathcal{B}[\![b]\!]\}$$

mit $w \equiv$ **while** (b) c Induktion über $i \geq 0$

$$\{(\sigma,\sigma') \mid \underbrace{(\sigma,\mathbf{1}) \in \mathcal{B}[\![b]\!]}_{\langle b,\sigma\rangle \to_{Bexp} \mathbf{1}}, \quad \underbrace{(\sigma,\sigma'') \in \mathcal{C}[\![c]\!]}_{(\mathit{strukt.\ IH})\langle c,\sigma\rangle \to_{Stmt} \sigma'} \quad , \quad \underbrace{(\sigma'',\sigma') \in \Gamma^i(\emptyset)}_{(\leq i\ IH)\langle w,\sigma''\rangle \to_{Stmt} \sigma'} \quad \cup\{(\sigma,\sigma) \mid \underbrace{(\sigma,\mathbf{0}) \in}_{\langle w,\sigma\rangle \to_{s}}$$

$$\frac{}{\langle w,\sigma\rangle \to_{Stmt} \sigma'}$$

Äquivalenz operationale und denotationale Semantik

- Für alle $c \in \mathbf{Stmt}$, für alle Zustände σ, σ':

$$\langle c, \sigma \rangle \to_{Stmt} \sigma' \Leftrightarrow (\sigma, \sigma') \in \mathcal{C}[\![c]\!]$$

$$\langle c, \sigma \rangle \to_{Stmt} \bot \Rightarrow \sigma \notin Dom(\mathcal{C}[\![c]\!])$$

- Gegenbeispiel für $\Leftarrow$ in der zweiten Aussage: wähle $c \equiv \mathit{while}(1)\{\}$: $\mathcal{C}[\![c]\!] = \emptyset$ aber $\langle c, \sigma \rangle \to_{Stmt} \bot$ gilt nicht (sondern?).

Fahrplan

- Einführung
- Operationale Semantik
- Denotationale Semantik
- Äquivalenz der Operationalen und Denotationalen Semantik
- Der Floyd-Hoare-Kalkül
- Invarianten und die Korrektheit des Floyd-Hoare-Kalküls
- Strukturierte Datentypen
- Verifikationsbedingungen
- Vorwärts mit Floyd und Hoare
- Modellierung
- Spezifikation von Funktionen
- Referenzen und Speichermodelle
- Funktionsaufrufe und das Framing-Problem
- Ausblick und Rückblick