Hoare Calculus

Andreas Podelski

May 17, 2017

Loop Invariant, Invariant, Inductive Invariant

given while command $C \equiv$ **while** b **do** C_0

- θ is *loop invariant* if:

$$\{\theta \wedge b\}\ C_0\ \{\theta\}$$

Loop Invariant, Invariant, Inductive Invariant

given while command $C \equiv$ **while** b **do** C_0

- θ is *loop invariant* if:

$$\{\theta \wedge b\}\ C_0\ \{\theta\}$$

- given precondition ϕ, θ is *invariant* if:

$$\{\phi\}\ \textbf{skip}\ \{\theta\}$$
$$\{\phi\}\ \textbf{if}\ b\ \textbf{then}\ C_0\ \textbf{else skip}\ \{\theta\}$$
$$\{\phi\}\ \textbf{if}\ b\ \textbf{then}\ \{C_0\ ;\ \textbf{if}\ b\ \textbf{then}\ C_0\ \textbf{else skip}\}\ \textbf{else skip}\ \{\theta\}$$
$$\ldots$$

Loop Invariant, Invariant, Inductive Invariant

given while command $C \equiv \textbf{while } b \textbf{ do } C_0$

- θ is *loop invariant* if:

$$\{\theta \wedge b\}\ C_0\ \{\theta\}$$

- given precondition ϕ, θ is *invariant* if:

$$\begin{array}{l}
\{\phi\}\ \textbf{skip}\ \{\theta\} \\
\{\phi\}\ \textbf{if } b \textbf{ then } C_0 \textbf{ else skip}\ \{\theta\} \\
\{\phi\}\ \textbf{if } b \textbf{ then } \{C_0\ ;\ \textbf{if } b \textbf{ then } C_0 \textbf{ else skip}\} \textbf{ else skip}\ \{\theta\} \\
\ldots
\end{array}$$

- given precondition ϕ, θ is *inductive invariant* if:

$$\begin{array}{l}
\{\phi\}\ \textbf{skip}\ \{\theta\} \\
\{\theta \wedge b\}\ C_0\ \{\theta\}
\end{array}$$

Annotated Programs

- expression (where f maps into **Val**)

$$e ::= x \mid f(e_1, \ldots, e_n)$$

Annotated Programs

- expression (where f maps into **Val**)

$$e \ ::= \ x \mid f(e_1, \ldots, e_n)$$

- Boolean expression (where f maps into $\{\mathbf{T}, \mathbf{F}\}$)

$$b \ ::= \ x \mid f(e_1, \ldots, e_n)$$

Annotated Programs

- expression (where f maps into **Val**)

$$e \ ::= \ x \mid f(e_1, \ldots, e_n)$$

- Boolean expression (where f maps into $\{\mathbf{T}, \mathbf{F}\}$)

$$b \ ::= \ x \mid f(e_1, \ldots, e_n)$$

- assertion

$$\phi, \psi, \theta \ ::= \ b \mid \top \mid \bot \mid \neg\phi \mid \phi \vee \psi \mid \exists x.\phi$$

Annotated Programs

- expression (where f maps into **Val**)

$$e \;::=\; x \mid f(e_1, \ldots, e_n)$$

- Boolean expression (where f maps into $\{\mathbf{T}, \mathbf{F}\}$)

$$b \;::=\; x \mid f(e_1, \ldots, e_n)$$

- assertion

$$\phi, \psi, \theta \;::=\; b \mid \top \mid \bot \mid \neg\phi \mid \phi \vee \psi \mid \exists x.\phi$$

- command

$$\begin{array}{rcl} C & ::= & \mathbf{skip} \mid x := e \mid C_1; C_2 \mid \mathbf{if}\ b\ \mathbf{then}\ C_1\ \mathbf{else}\ C_2 \mid \\ & & \mathbf{while}\ b\ \mathbf{do}\ \{\theta\}\ C \end{array}$$

Example: Factorial function

- $$
\begin{array}{l}
\{n \geq 0\} \\
f := 1; \\
i := 1; \\
\textbf{while } i \leq n \textbf{ do } \{f = fact(i-1) \wedge i \leq n+1\}\ \{ \\
\quad f := f \times i \\
\quad i := i + 1 \\
\} \\
\{f = fact(n)\}
\end{array}
$$

Example: Factorial function

- $$\begin{array}{l} \{n \geq 0\} \\ f := 1; \\ i := 1; \\ \textbf{while } i \leq n \textbf{ do } \{f = fact(i-1) \wedge i \leq n+1\} \; \{ \\ \quad f := f \times i \\ \quad i := i + 1 \\ \} \\ \{f = fact(n)\} \end{array}$$
- function symbol *fact* used in assertions ϕ, ψ, θ
  not used in commands C

Example: Factorial function

- $$\begin{array}{l} \{n \geq 0\} \\ f := 1; \\ i := 1; \\ \textbf{while } i \leq n \textbf{ do } \{f = fact(i-1) \wedge i \leq n+1\}\ \{ \\ \quad f := f \times i \\ \quad i := i + 1 \\ \} \\ \{f = fact(n)\} \end{array}$$
- function symbol *fact* used in assertions ϕ, ψ, θ
  not used in commands C
- interpretation of function symbol *fact* in logical model for integers (bounded or unbounded)

Example: Factorial function

- $$
\begin{array}{l}
\{n \geq 0\} \\
f := 1; \\
i := 1; \\
\textbf{while } i \leq n \textbf{ do } \{f = fact(i-1) \land i \leq n+1\} \ \{ \\
\qquad f := f \times i \\
\qquad i := i + 1 \\
\} \\
\{f = fact(n)\}
\end{array}
$$
- function symbol *fact* used in assertions ϕ, ψ, θ
  not used in commands C
- interpretation of function symbol *fact* in logical model for integers (bounded or unbounded)
- axioms added in set of assertions Γ

$$
\begin{array}{l}
fact(0) = 1 \\
\forall n.\ \ n > 0 \rightarrow fact(n) = n \times fact(n-1)
\end{array}
$$

Loop Unfolding

- equivalence (proved using the semantics of programs)

$$\textbf{while } b \textbf{ do } C_0 \;\equiv\; \textbf{if } b \textbf{ then } \{C_0 \;;\, \textbf{while } b \textbf{ do } C_0\} \textbf{ else skip}$$

Loop Unfolding

- equivalence (proved using the semantics of programs)

$$\textbf{while } b \textbf{ do } C_0 \;\equiv\; \textbf{if } b \textbf{ then } \{C_0 \,;\, \textbf{while } b \textbf{ do } C_0\} \textbf{ else skip}$$

- number of unfoldings may be huge

Loop Unfolding

- equivalence (proved using the semantics of programs)

$$\textbf{while } b \textbf{ do } C_0 \;\equiv\; \textbf{if } b \textbf{ then } \{C_0 \,;\, \textbf{while } b \textbf{ do } C_0\} \textbf{ else skip}$$

- number of unfoldings may be huge
- number of unfoldings *statically* not known

System $\mathcal{H}$ (1)

- Hoare triple $\{\phi\}\ C\ \{\psi\}$ derivable in $\mathcal{H}$ if exists a derivation using the axioms and inference rules of $\mathcal{H}$

System $\mathcal{H}$ (1)

- Hoare triple $\{\phi\}\ C\ \{\psi\}$ derivable in $\mathcal{H}$ if exists a derivation using the axioms and inference rules of $\mathcal{H}$
- skip

$$\frac{}{\{\phi\}\ \textbf{skip}\ \{\phi\}}$$

System $\mathcal{H}$ (1)

- Hoare triple $\{\phi\}\ C\ \{\psi\}$ derivable in $\mathcal{H}$ if exists a derivation using the axioms and inference rules of $\mathcal{H}$
- skip

$$\frac{}{\{\phi\}\ \textbf{skip}\ \{\phi\}}$$

- assignment

$$\frac{}{\{\psi[e/x]\}\ x := e\ \{\psi\}}$$

System $\mathcal{H}$ (2)

- sequential command $C \equiv C_1 \; ; \; C_2$

$$\frac{\{\phi\}\ C_1\ \{\phi'\} \qquad \{\phi'\}\ C\ \{\psi\}}{\{\phi\}\ C\ \{\psi\}}$$

System $\mathcal{H}$ (2)

- sequential command $C \;\equiv\; C_1 \,;\, C_2$

$$\frac{\{\phi\}\ C_1\ \{\phi'\} \qquad \{\phi'\}\ C\ \{\psi\}}{\{\phi\}\ C\ \{\psi\}}$$

- conditional command $C \;\equiv\; \mathbf{if}\ b\ \mathbf{then}\ C_1\ \mathbf{else}\ C_2$

$$\frac{\{\phi \wedge b\}\ C_1\ \{\psi\} \qquad \{\phi \wedge \neg b\}\ C\ \{\psi\}}{\{\phi\}\ C\ \{\psi\}}$$

System $\mathcal{H}$ (3)

- while command $C \equiv \textbf{while}\ b\ \textbf{do}\ \{\theta\}\ C_0$

$$\frac{\{\theta \wedge b\}\ C_0\ \{\theta\}}{\{\theta\}\ C\ \{\theta \wedge \neg b\}}$$

System $\mathcal{H}$ (3)

- while command $C \equiv$ **while** b **do** $\{\theta\}\ C_0$

$$\frac{\{\theta \wedge b\}\ C_0\ \{\theta\}}{\{\theta\}\ C\ \{\theta \wedge \neg b\}}$$

- strengthen precondition, weaken postcondition

$$\frac{\{\phi\}\ C\ \{\psi\}}{\{\phi'\}\ C\ \{\psi'\}} \quad \text{if} \quad \phi' \to \phi \quad \text{and} \quad \psi \to \psi'$$

System $\mathcal{H}$ (3)

- while command $C \equiv \textbf{while}\ b\ \textbf{do}\ \{\theta\}\ C_0$

$$\frac{\{\theta \wedge b\}\ C_0\ \{\theta\}}{\{\theta\}\ C\ \{\theta \wedge \neg b\}}$$

- strengthen precondition, weaken postcondition

$$\frac{\{\phi\}\ C\ \{\psi\}}{\{\phi'\}\ C\ \{\psi'\}} \quad \text{if} \quad \phi' \to \phi \quad \text{and} \quad \psi \to \psi'$$

System $\mathcal{H}$ (3)

- while command $C \equiv$ **while** b **do** $\{\theta\}\ C_0$

$$\frac{\{\theta \wedge b\}\ C_0\ \{\theta\}}{\{\theta\}\ C\ \{\theta \wedge \neg b\}}$$

- strengthen precondition, weaken postcondition

$$\frac{\{\phi\}\ C\ \{\psi\}}{\{\phi'\}\ C\ \{\psi'\}} \quad \text{if} \quad \phi' \to \phi \quad \text{and} \quad \psi \to \psi'$$

- Hoare triple derivable in all logicals models in which implications in side condition are valid

Soundness of $\mathcal{H}$

- if $\{\phi\}$ C $\{\psi\}$ derivable in given logical model
  then $\{\phi\}$ C $\{\psi\}$ valid in the model

Soundness of $\mathcal{H}$

- if $\{\phi\}$ C $\{\psi\}$ derivable in given logical model
  then $\{\phi\}$ C $\{\psi\}$ valid in the model
- if $\{\phi\}$ C $\{\psi\}$ derivable from given set of assertions Γ
  then $\{\phi\}$ C $\{\psi\}$ valid in all models in which Γ is valid

Soundness of $\mathcal{H}$

- if $\{\phi\}$ C $\{\psi\}$ derivable in given logical model
  then $\{\phi\}$ C $\{\psi\}$ valid in the model
- if $\{\phi\}$ C $\{\psi\}$ derivable from given set of assertions Γ
  then $\{\phi\}$ C $\{\psi\}$ valid in all models in which Γ is valid
- inverse does not hold in general

Soundness of $\mathcal{H}$

- if $\{\phi\}$ C $\{\psi\}$ derivable in given logical model
  then $\{\phi\}$ C $\{\psi\}$ valid in the model
- if $\{\phi\}$ C $\{\psi\}$ derivable from given set of assertions Γ
  then $\{\phi\}$ C $\{\psi\}$ valid in all models in which Γ is valid
- inverse does not hold in general
- derivability depends on annotation with loop invariants, validity does not

Example: Factorial function

$$
\begin{array}{l}
\{n \geq 0\} \\
f := 1; \\
i := 1; \\
\textbf{while } i \leq n \textbf{ do } \{f = fact(i-1) \wedge i \leq n+1\} \ \{ \\
\qquad f := f \times i \\
\qquad i := i + 1 \\
\} \\
\{f = fact(n)\}
\end{array}
$$

Adaptation

- $\{n = 10\}$ **Fact** $\{f = fact(n)\}$ valid

Adaptation

- $\{n = 10\}$ **Fact** $\{f = fact(n)\}$ valid
- derivable from $\{n \geq 0\}$ **Fact** $\{f = fact(n)\}$

Adaptation

- $\{n = 10\}$ **Fact** $\{f = fact(n)\}$ valid
- derivable from $\{n \geq 0\}$ **Fact** $\{f = fact(n)\}$
- not derivable from
  $\{n \geq 0 \wedge n = n_0\}$ **Fact** $\{f = fact(n) \wedge n = n_0\}$

Adaptation

- $\{n = 10\}$ **Fact** $\{f = fact(n)\}$ valid
- derivable from $\{n \geq 0\}$ **Fact** $\{f = fact(n)\}$
- not derivable from
  $\{n \geq 0 \wedge n = n_0\}$ **Fact** $\{f = fact(n) \wedge n = n_0\}$
  none of the implications in side conditions is valid

Adaptation

- $\{n = 10\}$ **Fact** $\{f = fact(n)\}$ valid
- derivable from $\{n \geq 0\}$ **Fact** $\{f = fact(n)\}$
- not derivable from
  $\{n \geq 0 \wedge n = n_0\}$ **Fact** $\{f = fact(n) \wedge n = n_0\}$

  none of the implications in side conditions is valid
- more complicated inference rule for 'instantiating a Hoare triple' with auxiliary variables

Adaptation

- $\{n = 10\}$ **Fact** $\{f = fact(n)\}$ valid
- derivable from $\{n \geq 0\}$ **Fact** $\{f = fact(n)\}$
- not derivable from
  $\{n \geq 0 \wedge n = n_0\}$ **Fact** $\{f = fact(n) \wedge n = n_0\}$
  none of the implications in side conditions is valid
- more complicated inference rule for 'instantiating a Hoare triple' with auxiliary variables
- in practice, we will need adaptation only for *procedure contracts*
  which we will introduce later