Introduction to Formal Methods

Lecture 6

Hoare Logic

Hossein Hojjat & Fatemeh Ghassemi

October 9, 2018

Loop-Free Programs as Relations: Summary

command c	$R(c)$	$\rho(c)$
$(x = t)$	$x' = t \wedge \bigwedge_{v \in V \setminus \{x\}} v' = v$	
$c_1; c_2$	$\exists \vec{z}. R(c_1)[\vec{x'} := \vec{z}] \wedge R(c_2)[\vec{x} := \vec{z}]$	$\rho(c_1) \circ \rho(c_2)$
$c_1 \;[]\; c_2$	$R(c_1) \vee R(c_2)$	$\rho(c_1) \cup \rho(c_2)$
$\texttt{assume}(F)$	$F \wedge \bigwedge_{v \in V} v' = v$	$\Delta_{S(F)}$

Putting Conditions on Sets Makes them Smaller

- Let P_1 and P_2 be formulas ("conditions") whose free variables are among $\vec{x}$
  (Those variables may denote program state)
- When we say "condition P_1 is stronger than condition P_2" it simply means

$$\forall \vec{x}.(P_1 \rightarrow P_2)$$

  - if we know P_1, we immediately get (conclude) P_2
  - if we know P_2, we need not be able to conclude P_1
- Stronger condition = smaller set: if P_1 is stronger than P_2 then

$$\{\vec{x} \mid P_1\} \subseteq \{\vec{x} \mid P_2\}$$

- Strongest possible condition: "false" $\equiv$ smallest set: $\emptyset$
- Weakest condition: "true" $\equiv$ biggest set: set of all tuples

About Hoare Logic

- We have seen how to translate programs into relations
- We will use these relations in a proof system called Hoare logic
- Hoare logic is a way of inserting annotations into code to make proofs about (imperative) program behavior simpler

Example proof:

```
// {0 <= y}
i = y;
// {0 <= y & i = y}
r = 0;
// {0 < = y & i = y & r = 0}
while // {r = (y − i) ∗ x & 0 <= i}
  (i > 0) {
     // {r = (y − i) ∗ x & 0 < i}
     r = r + x;
     // {r = (y − i + 1) ∗ x & 0 < i}
     i = i − 1;
     // {r = (y − i) ∗ x & 0 <= i}
}
// { r = x ∗ y}
```

Hoare Triple

$P, Q \subseteq S \qquad r \subseteq S \times S$

Hoare Triple:

$$\{P\}\ r\ \{Q\} \iff \forall s, s' \in S.(s \in P \land (s, s') \in r \rightarrow s' \in Q)$$

$\{P\}$ does not denote a singleton set containing P but is just a notation for an "assertion" around a command. Likewise for $\{Q\}$

Strongest postcondition:

$$sp(P, r) = \{s' \mid \exists s. s \in P \land (s, s') \in r\}$$

Weakest precondition:

$$wp(r, Q) = \{s \mid \forall s'.(s, s') \in r \rightarrow s' \in Q\}$$

Sir Tony Hoare

Sir Charles Antony Richard Hoare giving a conference at EPFL on 20 June 2011

Born Charles Antony Richard Hoare

Exercise

Which Hoare triples are valid?

1. {j = a} j := j + 1 {a = j + 1}
2. {i = j} i := j + i {i > j}
3. {j = a + b} i := b; j := a {j = 2 * a}
4. {i > j} j := i+1; i := j+1 {i > j}
5. {i != j} `if` i > j `then` m := i - j `else` m := j - i {m > 0}
6. {i = 3 * j} `if` i > j `then` m := i - j `else` m := j - i {m - 2 * j = 0}

Postconditions and Their Strength

What is the relationship between these postconditions?

$\{x = 5\}$ x := x + 2 $\{x > 0\}$

$\{x = 5\}$ x := x + 2 $\{x = 7\}$

- weakest conditions (predicates) correspond to largest sets
- strongest conditions (predicates) correspond to smallest sets

that satisfy a given property

(Graphically, a stronger condition $x > 0 \wedge y > 0$ denotes one quadrant in plane,
whereas a weaker condition $x > 0$ denotes the entire half-plane.)

Strongest Postconditions

- Some valid Hoare Triples

$$\begin{array}{lll} \{x = 5\} & x := x + 5 & \{\text{true}\} \\ \{x = 5\} & x := x + 5 & \{x > 0\} \\ \{x = 5\} & x := x + 5 & \{x = 10 \lor x = 5\} \\ \{x = 5\} & x := x + 5 & \{x = 10\} \end{array}$$

- All are valid but $x = 10$ is the most useful one
  - Strongest postcondition
- If $\{P\}\ r\ \{Q\}$ and for all Q' such that $\{P\}\ r\ \{Q'\}$, $Q \to Q'$, then Q is the strongest postcondition of r with respect to P
- check: $x = 10 \to \text{true}$
- check: $x = 10 \to x > 0$
- check: $x = 10 \to x = 10 \lor x = 5$
- check: $x = 10 \to x = 10$

Weakest Preconditions

- Some valid Hoare Triples (assume an extension of IMP with division)

$$\begin{array}{lll} \{x = 5 \land y = 10\} & z := x/y & \{z < 1\} \\ \{x < y \land y > 0\} & z := x/y & \{z < 1\} \\ \{y \neq 0 \land x/y < 1\} & z := x/y & \{z < 1\} \end{array}$$

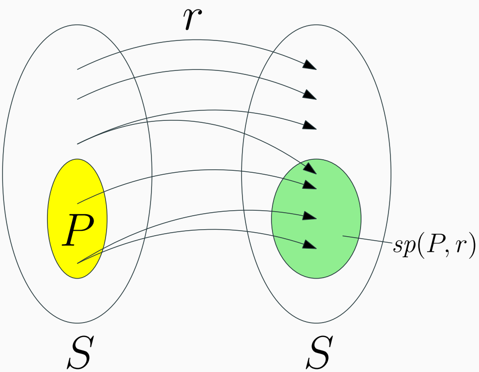
- All are valid but $y \neq 0 \land x/y < 1$ is the most useful one
- It allows us to invoke the program in the most general condition
  - Weakest precondition
- If $\{P\}\ r\ \{Q\}$ and for all P' such that $\{P'\}\ r\ \{Q\}$, $P' \to P$, then P is the weakest precondition of r with respect to Q

Strongest Postcondition

Definition: For $P \subseteq S$, $r \subseteq S \times S$,

$$sp(P, r) = \{s' \mid \exists s. s \in P \wedge (s.s') \in r\}$$

This is simply the relation image of a set

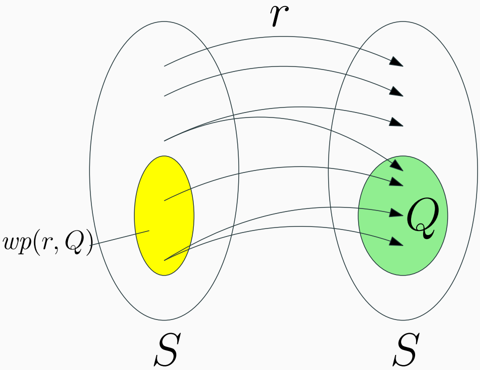


Weakest Precondition

Definition: For $P \subseteq S$, $r \subseteq S \times S$,

$$\textit{wp}(r, Q) = \{s \mid \forall s'.(s.s') \in r \rightarrow s' \in Q\}$$

Note that this is in general not the same as $\textit{sp}(Q, r^{-1})$ when the relation is non-deterministic or partial



Three Forms of Hoare Triple

Lemma: the following three conditions are equivalent:

- $\{P\}\ r\{Q\}$
- $P \subseteq \textit{wp}(r, Q)$
- $\textit{sp}(P, r) \subseteq Q$

Three Forms of Hoare Triple

Lemma: the following three conditions are equivalent:

- $\{P\}\ r\{Q\}$
- $P \subseteq \textit{wp}(r, Q)$
- $\textit{sp}(P, r) \subseteq Q$

Proof. The three conditions expand into the following three formulas

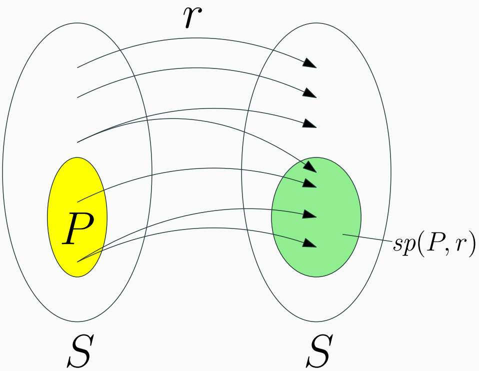
- $\forall s, s'.\big((s \in P \land (s, s') \in r) \to s' \in Q\big)$
- $\forall s.\big(s \in P \to (\forall s'.(s, s') \in r \to s' \in Q)\big)$
- $\forall s'.\big((\exists s.s \in P \land (s, s') \in P) \to s' \in Q\big)$

which are easy to show equivalent using basic first-order logic properties

Lemma: Characterization of sp

$sp(P, r)$ is the the smallest set Q such that $\{P\}\ r\ \{Q\}$, that is:

- $\{P\}\ r\ \{sp(P, r)\}$
- $\forall Q \subseteq S.\{P\}\ r\ \{Q\} \to sp(P, r) \subseteq Q$



$$\{P\}\ r\ \{Q\} \Leftrightarrow \forall s, s' \in S.(s \in P \land (s, s') \in r \to s' \in Q)$$
$$sp(P, r) = \{s' \mid \exists s.s \in P \land (s, s') \in r\}$$

Proof of Lemma: Characterization of sp

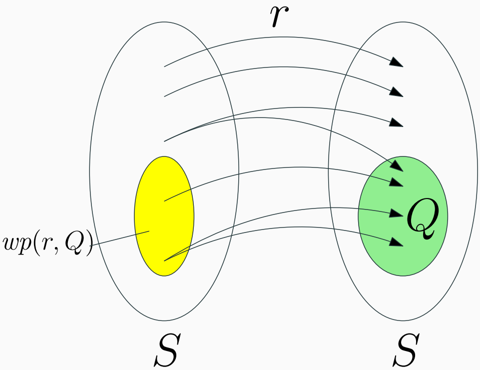
Apply Three Forms of Hoare triple. The two conditions then reduce to:

- $\mathit{sp}(P, r) \subseteq \mathit{sp}(P, r)$
- $\forall P \subseteq S. \mathit{sp}(P, r) \subseteq Q \to \mathit{sp}(P, r) \subseteq Q$

Lemma: Characterization of wp

$\mathit{wp}(r, Q)$ is the largest set P such that $\{P\}\ r\ \{Q\}$, that is:

- $\{\mathit{wp}(r, Q)\}\ r\ \{Q\}$
- $\forall P \subseteq S.\{P\}\ r\ \{Q\} \rightarrow P \subseteq \mathit{wp}(r, Q)$



$$\{P\}\ r\ \{Q\} \Leftrightarrow \forall s, s' \in S.(s \in P \wedge (s, s') \in r \rightarrow s' \in Q)$$
$$\mathit{wp}(r, Q) = \{s \mid \forall s'.(s, s') \in r \rightarrow s' \in Q\}$$

Exercise: Postcondition of inverse versus wp

Lemma:

$$S \backslash \mathit{wp}(r, Q) = \mathit{sp}(S \backslash Q, r^{-1})$$

In other words, when instead of good states we look at the

completement set of "error states", then *wp* corresponds to doing *sp* backwards.

Note that $r^{-1} = \{(y, x) \mid (x, y) \in r\}$ and is always defined

More Laws on Preconditions and Postconditions

Disjunctivity of sp

$$sp(P_1 \cup P_2, r) = sp(P_1, r) \cup sp(P_2, r)$$
$$sp(P, r_1 \cup r_2) = sp(P, r_1) \cup sp(P, r_2)$$

Conjunctivity of wp

$$wp(r, Q_1 \cap Q_2) = wp(r, Q_1) \cap wp(r, Q_2)$$
$$wp(r_1 \cup r_2, Q) = wp(r1, Q) \cap wp(r_2, Q)$$

Pointwise wp

$$wp(r, Q) = \{s \mid s \in S \wedge sp(\{s\}, r) \subseteq Q\}$$

Pointwise sp

$$sp(P, r) = \bigcup_{s \in P} sp(\{s\}, r)$$