CS 512 Lecture Notes

Nathan Cordner

22 March 2018

Today's class notes are largely based on the lecture notes *Hoare Logic and Variations* [HL+V].

1 Review of Formal Semantics of Classical Hoare Logic (HL)

We quickly recapped the semantics defined in Section 1.3 at the top of page 7 in [HL+V]. We noted that in the assignment operator, the notation $\sigma[x \mapsto n]$ assigns the value n to variable x. For example, passing in a tuple $\langle x, y, z\rangle = \langle 26, 5, 2\rangle$ to $\sigma[x \mapsto 15]$ produces $\langle 15, 5, 2\rangle$.

We also reviewed the idea of composing two relations together for the command $C_1; C_2$. For example, if

$$\begin{aligned} R &= \{(2,3),(4,5)\} \\ S &= \{(1,2),(3,4),(100,-6)\}, \end{aligned}$$

then working from S to R yields $R \circ S = \{(1,3),(3,5)\}$.

1.1 Solving Fixpoint Equations

In general, given a function $f(x)$ we call the problem $x = f(x)$ a *fixpoint* equation. It is not always easy to solve this problem analytically, so we often resort to using an iterative algorithm to repeatedly update an initial guess x_0 to obtain better solutions x_n that more closely satisfy $x_n = f(x_n)$. (For example, think of using Newton's method from calculus to find the solution to $f(x) - x = 0$).

It turns out that the key to the semantics of the `while-do` loop is a solution to a fixpoint equation. We have the relation

$$R = \{(\sigma,\sigma') \mid [\![B]\!]\sigma = \texttt{true} \text{ and } (\sigma,\sigma') \in R \circ [\![C]\!]_{\text{rel}}\} \cup \{(\sigma,\sigma) \mid [\![B]\!]\sigma = \texttt{false}\},$$

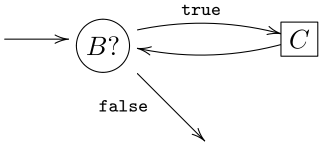
and the function $\mathcal{F}$ such that

$$\mathcal{F}(R) \triangleq \{(\sigma,\sigma') \mid [\![B]\!]\sigma = \texttt{true} \text{ and } (\sigma,\sigma') \in R \circ [\![C]\!]_{\text{rel}}\} \cup \{(\sigma,\sigma) \mid [\![B]\!]\sigma = \texttt{false}\}.$$

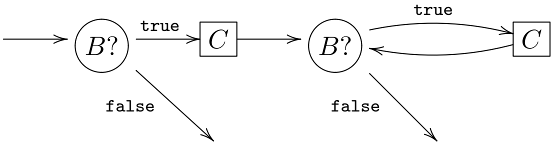
Thus defining the semantics of the `while-do` loop relies on solving the fixpoint equation $\mathcal{F}(R) = R$. Let's look at a picture. Suppose we have X_i, where i denotes the number of times we have "unwrapped" our `while-do` loop. Write $X_0 = \emptyset$. Then

$$\mathcal{F}(X_0) = \{(\sigma,\sigma) \mid [\![B]\!]\sigma = \texttt{false}\}.$$

Set $X_1 = \mathcal{F}(X_0)$, $X_2 = \mathcal{F}(X_1)$, and so on. Since X_0 is empty, there is no picture to draw. For X_1, we have the diagram



For X_2, the picture looks like



In general, there would be i copies of C in the diagram for X_i. We see that $X_0 \subseteq X_1 \subseteq X_2 \subseteq \dots$.

Sometimes fixpoint equations yield many solutions. We briefly considered Example 6 on page 8 of [HL+V]. We define the set

$$A \triangleq \{\langle n, 1 + (n \text{ div } 2) \cdot 2\rangle \mid n \in \mathbb{Z} \text{ and } n \geq 0\} \subseteq \mathbb{Z} \times \mathbb{Z},$$

and, for some relation $R \subseteq \mathbb{Z} \times \mathbb{Z}$, we solve a fixpoint equation like

$$R = \mathcal{F}(R) \text{ where } \mathcal{F}(R) \triangleq A \cup (R \text{ div } \langle 2, 1\rangle) \cdot \langle 2, 1\rangle.$$

It turns out the solution set is infinite. In such cases, it is best to choose the *least* fixpoint (according to whatever our notion of size is for the particular solutions at hand).

We also briefly touched on Example 7 on page 9 of [HL+V] concerning the factorial function as another example of solving fixpoint equations.

2 Relational Hoare Logic (RHL)

This section begins in Section 2 on page 12 in [HL+V]. Recall that classical Hoare Logic is denoted by *Hoare triples* that look like

$$\{\phi\}P\{\psi\},$$

where ϕ is a *precondition* and ψ is a *postcondition.* By contrast, the relational Hoare Logic is denoted by *Hoare quadruples* that look like

$$\{\Phi\}C_1 \sim C_2\{\Psi\},$$

where Φ is a *prerelation* and Ψ is a *postrelation.* This notation is used to show that C_1 and C_2 are related to each other. We note that when started on Φ-related states, either C_1 and C_2 both diverge, or they both terminate in Ψ-related states.

The relational Hoare Logic also has some inference rules. A (currently) incomplete list appears on page 15 of [HL+V].

We briefly commented on Example 12 on page 13 of [HL+V] as a way to use relational Hoare Logic to show how two different programs can be related to each other. Here P_1 and P_2 are written in distinct variables, but we can show that they are related by using a Hoare quadruple.

3 Probabilistic Hoare Logic (pHL)

The notes on probabilistic Hoare Logic start in Section 4 on page 17 of [HL+V]. The main distinction between pHL and HL is the introduction of randomized steps in the program. For example, we can write

$$C \triangleq \big((x := 1) \oplus_{1/4} (x := 5)\big).$$

We interpret this notation as x having a $\frac{1}{4}$ chance of being assigned to be 1, and a $\frac{3}{4}$ chance of being assigned to 5.

We note that "randomized" is not the same as "nondeterministic." The coin flip program presented in Example 15 on page 17 of [HL+V] shows this difference. Even with the randomized step, the program will always terminate. However, one possible path available to nondeterminism is "heads" forever which would mean the program never terminates.

In Example 16 on page 18 of [HL+V], we adapt some classical HL into pHL. Suppose C is the instruction $y := x + 2$. We can write the Hoare triple

$$\{x = 1\}C\{y = 3\}.$$

But suppose we introduce a randomized step, and create the instruction C' given by $y := x + \texttt{random}(2)$. We need to tweak the pre- and post-conditions, and can now write something like

$$\{\mathbb{P}(x = 1) \geq 3/4\}C'\{\mathbb{P}(y = 3) \geq 1/4\}.$$