

SOUNDNESS OF THE AXIOMATIC RULES

Any rule-based system, if it is to be used successfully, should be *sound*. Soundness is intimately related to the idea of *validity*. A valid expression in the language of the system is one which is true based on some set of assumptions, which might be empty. In our logical language of assertions, there is clearly a relationship between the store, as defined in operational semantics, and the form itself. Instead of writing $[s, \sigma] \rightarrow v$ as we did in that semantics, where s is *syntactic* form, we will write $\sigma \vDash a$, where a is an expression in the assertion language. We will read it as ‘ a is valid in σ ’, or ‘ a follows from σ ’, or ‘ a is satisfied by σ ’. We would like to show that all possible partial correctness assertions are valid, but only those which are provable, i.e. those for which a derivation exists, using the axiomatic rules. If we can prove this, then we can say that the rules are sound. Another way of looking at it is that no derivation that uses the rules produces a partial completeness assertion that is false. In symbols it is

$$\vdash \{A\}c\{B\} \Rightarrow \vDash \{A\}c\{B\}$$

Or, anything that is proveable is also valid.

First, we shall be accurate about our assertion language. Starting with the language of arithmetic expressions, we will add integer variables so that we can make assertions about them. These are distinct from the identifiers. The semantics of this language is given by operational semantics:

$$[n, \sigma] \rightarrow n$$

$$[I, \sigma] \rightarrow \sigma(I)$$

$$[i, \sigma] \rightarrow I(i) \quad I : i \rightarrow \mathbb{Z} \text{ is an interpretation function}$$

$$[a_1 + a_2, \sigma] \rightarrow n_1 + n_2 \text{ where } [a_1, \sigma] \rightarrow n_1 \text{ and } [a_2, \sigma] \rightarrow n_2$$

And similarly for the other arithmetic operators. To this we add the relational operators and the logical operators to make the assertion language. Its semantics are given by those stores that satisfy an assertion (using the operational rule forms):

$$\sigma \vDash \text{true}$$

$$\sigma \vDash (a_1 + a_2) \text{ if } [a_1, \sigma] \rightarrow n_1, [a_2, \sigma] \rightarrow n_2, \text{ and } n_1 = n_2$$

$$\sigma \vDash (a_1 \leq a_2) \text{ if } [a_1, \sigma] \rightarrow n_1, [a_2, \sigma] \rightarrow n_2, \text{ and } n_1 \leq n_2$$

$$\sigma \vDash A \wedge B \text{ if } \sigma \vDash A, \text{ and } \sigma \vDash B$$

$$\sigma \vDash A \vee B \text{ if } \sigma \vDash A, \text{ or } \sigma \vDash B$$

$$\sigma \vDash \sim A \text{ if } \text{not } \sigma \vDash A$$

$$\sigma \vDash A \Rightarrow B \text{ if } \text{not } \sigma \vDash A, \text{ or } \sigma \vDash B$$

$$\sigma \vDash \forall i. A \text{ if } \sigma \vDash A \text{ for all possible } i \in \mathbb{Z}$$

$$\sigma \vDash \exists i. A \text{ if } \sigma \vDash A \text{ for some } i \in \mathbb{Z}$$

Actually we should qualify all these validity assertions with the interpretation function I which gives meaning to the integer variables.

The partial correctness assertion in these terms is

$$\sigma \vDash \{A\}c\{B\} \text{ iff } \sigma \vDash A \wedge [c, \sigma] \rightarrow \sigma' \Rightarrow \sigma' \vDash B$$

Actually we are interested in whether $\vDash \{A\}c\{B\}$ for all stores and all interpretations I .

THE PROOF OF SOUNDNESS

The proof relies on a particular fact about substitution and updated stores:

$$\text{If } [a_0[a \setminus X], \sigma] \rightarrow n_1 \text{ and } [a_0, \sigma[X \mapsto n]] \rightarrow n_2 \text{ where } [a, \sigma] \rightarrow n, \text{ then } n_1 = n_2$$

We can prove this by using structural induction on the possible forms of a_0 .

$$\text{If } a \equiv n, \text{ clearly } [n[a \setminus X], \sigma] \rightarrow n, \text{ and } [n, \sigma[n \mapsto X]] \rightarrow n$$

If $a \equiv I$, and $I \neq X$, $[I[a \setminus X], \sigma] \rightarrow \sigma(I)$, and $[I, \sigma[X \mapsto n]] \rightarrow \sigma(I)$, where $[a, \sigma] \rightarrow n$

If $a \equiv I$, and $I = X$, then $[X[a \setminus X], \sigma] \rightarrow n$, and $[X, \sigma[X \mapsto n]] \rightarrow n$, where $[a, \sigma] \rightarrow n$

If $a = i$, then $[i[a \setminus X], \sigma] \rightarrow I(i)$, and $[i, \sigma[X \mapsto n]] \rightarrow I(i)$

If $a \equiv a_1 + a_2$, assume the fact is true of a_1 and a_2 and prove it true of $a_1 + a_2$. i.e. assume that

$[a_1[a \setminus X], \sigma] \rightarrow n_1$, and $[a_1, \sigma[X \mapsto n]] \rightarrow n_1$, and also $[a_2[a \setminus X], \sigma] \rightarrow n_2$, and $[a_2, \sigma[X \mapsto n]] \rightarrow n_2$, both where $[a, \sigma] \rightarrow n$.

But from the operational rule, $[a_1[a \setminus X] + a_2[a \setminus X], \sigma] \rightarrow n_1 + n_2$, i.e. $[(a_1 + a_2)[a \setminus X], \sigma] \rightarrow n_1 + n_2$. Also $[a_1 + a_2, \sigma[X \mapsto n]] \rightarrow n_1 + n_2$, so the fact is true of $a_1 + a_2$.

Since we have proved it for all forms, it is true of any expression.

We can prove a similar fact about valid assertions also by using structural induction:

$\sigma \vDash B[a \setminus X]$ iff $\sigma[X \mapsto n] \vDash B$, where $[a, \sigma] \rightarrow n$.

The proof of soundness uses these facts just proved. We shall use structural induction as we did for determinism of expressions in the operational semantics, with well-founded rule induction for the loop form. We will prove

$\vdash \{A\}c\{B\} \Rightarrow \vDash \{A\}c\{B\}$ for all commands by examining each possible command.

If $c \equiv \text{nop}$, clearly $\vDash \{A\}\text{nop}\{A\}$.

If $c \equiv X = a$, since the axiomatic rule is $\{B[a \setminus X]\}X = a\{B\}$, we need to show that $\sigma \vDash B[a \setminus X] \Rightarrow \sigma' \vDash B$, where $[X = a, \sigma] \rightarrow \sigma'$. From the substitution fact, we know that $\sigma \vDash B[a \setminus X] \Rightarrow \sigma[X \mapsto n] \vDash B$, where $[a, \sigma] \mapsto n$, and from the operational rule for assignment, we know that if $[a, \sigma] \rightarrow n$, then $[X = a, \sigma] \rightarrow \sigma[X \mapsto n]$, and so $\sigma' = \sigma[X \mapsto n]$, and $\sigma \vDash B[a \setminus X] \Rightarrow \sigma' \vDash B$, i.e. the rule is sound.

If $c = c_1; c_2$, assume that $\vDash \{A\}c_1\{C\}$, and $\vDash \{C\}c_2\{B\}$. Let $\sigma \vDash A$, then $\sigma_1 \vDash C$, where $[c_1, \sigma] \rightarrow \sigma_1$. Also $\sigma_2 \vDash B$, where $[c_2, \sigma_1] \rightarrow \sigma_2$. But from the operational rule, $[c_1; c_2, \sigma] \rightarrow \sigma_2$, so $\vDash \{A\}c_1; c_2\{B\}$.

If $c \equiv \text{if } b \text{ then } c_1 \text{ else } c_2$, assume $\vDash \{A \wedge b\}c_1\{B\}$ and $\vDash \{A \wedge \sim b\}c_2\{B\}$. Let $\sigma \vDash A$. Either $\sigma \vDash b$, or $\sigma \vDash \sim b$. If b is true, $\sigma \vDash A \wedge b$, so $\sigma' \vDash B$, where $[c_1, \sigma] \rightarrow \sigma'$. If b is false, $\sigma \vDash A \wedge \sim b$, and $\sigma'' \vDash B$, where $[c_2, \sigma] \rightarrow \sigma''$. But considering the operational rule, either $\sigma \vDash b$, so $[b, \sigma] \rightarrow \text{true}$, or $\sigma \vDash \sim b$, so $[b, \sigma] \rightarrow \text{false}$. Therefore $[\text{if } b \text{ then } c_1 \text{ else } c_2, \sigma] \rightarrow \sigma'$ when b true and $[\text{if } b \text{ then } c_1 \text{ else } c_2, \sigma] \rightarrow \sigma''$ when b is false. Thus $\vDash \{A\}\text{if } b \text{ then } c_1 \text{ else } c_2\{B\}$.

If $c \equiv \text{while } b \text{ do } c$, assume the invariant A where $\{A \wedge b\}c\{A\}$. We will use well-founded rule induction on the form of the operational while rule to derive $\{A\}\text{while } b \text{ do } c\{A \wedge \sim b\}$. The hypothesis is

$P(d) \Leftrightarrow d \vdash [\text{while } b \text{ do } c, \sigma] \rightarrow \sigma' \wedge \sigma \vDash A \Rightarrow \sigma' \vDash A \wedge \sim b$

We will show that $\forall d' \prec d. P(d') \Rightarrow P(d)$. Assume that $\forall d' \prec d. P(d')$.

When $[b, \sigma] \rightarrow \text{false}$, d has the form $\frac{\vdots}{\frac{[b, \sigma] \rightarrow \text{false}}{[\text{while } b \text{ do } c, \sigma] \rightarrow \sigma}}$, then $\sigma \vDash A \wedge \sim b$, and since $\sigma = \sigma'$, $\sigma' \vDash A \wedge \sim b$.

When $[b, \sigma] \rightarrow \text{true}$, d has the form $\frac{\vdots \quad \vdots \quad \vdots}{\frac{[b, \sigma] \rightarrow \text{false} \quad [c, \sigma] \rightarrow \sigma'' \quad [\text{while } b \text{ do } c, \sigma''] \rightarrow \sigma'}{[\text{while } b \text{ do } c, \sigma] \rightarrow \sigma'}}$

Let the subderivation of d for the recursive unfolding of the loop be d' . Since $d' \prec d, P(d')$, so $\sigma'' \vDash A \Rightarrow \sigma' \vDash A \wedge \sim b$. But $\{A \wedge b\}c\{A\}$, so $\sigma \vDash A \wedge b$, and $\sigma'' \vDash A$ where $[c, \sigma] \rightarrow \sigma''$. Therefore $\sigma' \vDash A \wedge \sim b$. Also $\sigma \vDash A$ since b is true, so $P(d)$ is true. The hypothesis is therefore true and so is $\{A\}\text{while } b \text{ do } c\{A \wedge \sim b\}$.

We have proved the soundness of using the operational rules for all command forms, therefore the system of axiomatic rules is sound.