Hoare Logic and Verification Condition Generation

Continued

David Pereira José Proença Eduardo Tovar

FVOCA 2021/2022

Formal Verification of Critical Applications

CISTER – ISEP
Porto, Portugal

https://cister-labs.github.io/fvoca2122

Verification Conditions Generation

How do these VCGen work?

- A VCGen algorithm takes as input a Hoare Triple $\{P\}\ C\ \{Q\}$ and returns a set of first-order logic proof obligations.
- The proof obligations represent side conditions of the form $F_1 \rightarrow F_2$

Algorithm for weakest precondition generation

$$
\begin{aligned}
wprec(\text{skip}, Q) &= Q \\
wprec(x := E, Q) &= Q[x \mapsto E] \\
wprec(C_1; C_2, Q) &= wprec(C_1, wprec(C_2, Q)) \\
wprec(\text{if } B \text{ then } C_1 \text{ else } C_2, Q) &= (B \rightarrow wprec(C_1, Q)) \wedge (\neg B \rightarrow wprec(C_2, Q)) \\
wprec(\text{while } B \text{ do } \{I\}\ C, Q) &= I
\end{aligned}
$$

Algorithm for generating Verification Conditions

$$VC(\{P\}\,\mathsf{skip}\{Q\}) = \{P \to Q\}$$

$$VC(\{P\}\, x := E\,\{Q\}) = \{P \to Q[x \mapsto E]\}$$

$$VC(\{P\}\, C_1; C_2\,\{Q\}) =$$

$$VC(\{P\}\, C_1\,\{wprec(C_2, Q)\}) \cup VC(\{wprec(C_2, Q)\}\, C_2\,\{Q\})$$

$$VC(\{P\}\,\mathsf{if}\,B\,\mathsf{then}\,C_1\,\mathsf{else}\,C_2\,\{Q\}) =$$

$$VC(\{P \wedge B\}\, C_1\{Q\}) \cup VC(\{P \wedge \neg B\}\, C_2\,\{Q\})$$

$$VC(\{P\}\,\mathsf{while}\,B\,\mathsf{do}\,\{I\}\,C\,\{Q\}) =$$

$$\{P \to I, I \wedge \neg B \to Q\} \cup VC(\{P \wedge \neg B\}\, C\,\{Q\})$$

Improved Verification Condition Generation

$$
\begin{aligned}
VC(\textsf{skip}, Q) &= \emptyset \\
VC(x := e, Q) &= \emptyset \\
VC(C_1; C_2, Q) &= VC(C_1, \mathit{wprec}(C_2, Q)) \cup VC(C_2, Q) \\
VC(\textsf{if } B \textsf{ then } C_1 \textsf{ else } C_2, Q) &= VC(C_1, Q) \cup VC(C_2, Q) \\
VC(\textsf{while } B \textsf{ do } \{I\}\ C, Q) &= \{(I \land B) \rightarrow \mathit{wprec}(C, I)\} \cup VC(C, I) \\
&\quad \{(I \land \neg B) \rightarrow Q\}
\end{aligned}
$$

$$
VCG(\{P\}\ C\ \{Q\}) = \{P \rightarrow \mathit{wprec}(C, Q)\} \cup VC(C, Q)
$$

Lets do some exercises

Consider the following program:

$$r := 1;\ i := 0;\ \textcolor{green}{\text{while}}\,(i < m)\{r = n^i \wedge 0 \leq i < m \wedge n > 0\}\,\textcolor{green}{\text{do}}\,\{r := r * n;\ i := i + 1;\}$$

Calculate the weakest precondition considering as post condition the following: $\{r = n^m\}$.

Next, calculate the VC assuming the precondition $n > 0 \wedge m \geq 0$

Finally, use the optimized algorithm for VC generation

Running wprec

$$\textbf{wprec}(r := 1; i := 0;\ WHILE, r = n^m) =$$
$$\textbf{wprec}(r := 1, \textbf{wprec}(i := 0; WHILE, r = n^m)) =$$
$$\textbf{wprec}(r := 1, \textbf{wprec}(i := 0, \textbf{wprec}(WHILE, r = n^m)))$$

Looking now only at part of the while loop. The rule for the wprec in the case of loops is to return the associated invariant, hence we have that:

$$\textbf{wprec}(\text{while}\,(i < m)\{r = n^i \land 0 \le i \le m \land n > 0\}\,\text{do}\,\{r := r * n;\ i := i + 1;\}, r = n^m) =$$
$$\{r = n^i \land 0 \le i \le m \land n > 0\}$$

Running wprec

Continuing:

$$\begin{aligned}
\textbf{wprec}(r := 1, \textbf{wprec}(i := 0, r = n^i \wedge 0 \leq i \leq m \wedge n > 0)) &= \\
\textbf{wprec}(r := 1, r = n^i \wedge 0 \leq i \leq m \wedge n > 0\,[i \mapsto 0]) &= \\
\textbf{wprec}(r := 1, r = n^0 \wedge 0 \leq 0 < m \wedge n > 0) &= \\
(r = n^0 \wedge 0 \leq 0 < m \wedge n > 0)\,[r \mapsto 1] &= \\
1 = n^0 \wedge 0 \leq 0 < m \wedge n > 0 &
\end{aligned}$$

Since we were considering $n > 0 \wedge m \geq 0$ as the precondition, it is straightforward to conclude that this precondition implies the weakest precondition generated.

Running VC

We will now apply the VC algorithm to generate all proof obligations.

$$
\begin{gathered}
\mathbf{VC}(\{n > 0 \land m \geq 0\}\ r := 1; i := 0;\ WHILE, \{r = n^m\}) \\
= \\
\mathbf{VC}(\{n > 0 \land m \geq 0\}\ r := 1\ \{wprec(i := 0;\ WHILE, \{r = n^m\})\}) \qquad (1) \\
\cup \\
\mathbf{VC}(\{\mathbf{wprec}(i := 0;\ WHILE, \{r = n^m\})\}\ i := 0;\ WHILE\ \{r = n^m\}) \qquad (2)
\end{gathered}
$$

Given the above expansion of the VC function, we should proceed by first reducing the weakest precondition generation, that is:

$$
\begin{array}{rc}
\mathbf{wprec}(i := 0;\ WHILE, \{r = n^m\}) & = \\
\mathbf{wprec}(i := 0, \mathbf{wprec}(WHILE, r = n^m)) & = \\
\mathbf{wprec}(i := 0, r = n^i \land 0 \leq i \leq m \land n > 0) & = \\
(r = n^i \land 0 \leq i \leq m \land n > 0)[i \mapsto 0] & = \\
r = n^0 \land 0 \leq 0 \leq m \land n > 0 &
\end{array}
$$

Running VC

So first, let us replace the *wprec* calls in our previous slide:.

$$\textbf{VC}(\{n > 0 \land m \geq 0\}\ r := 1\ \{r = n^0 \land 0 \leq 0 \leq m \land n > 0\}) \qquad (1)$$

$$\cup$$

$$\textbf{VC}(\{r = n^0 \land 0 \leq 0 \leq m \land n > 0\}\ i := 0;\ \mathit{WHILE}\ \{r = n^m\}) \qquad (2)$$

From (1) we can reach a final VC:

$$\textbf{VC}(\{n > 0 \land m \geq 0\}\ r := 1\ \{r = n^0 \land 0 \leq 0 \leq m \land n > 0\}) \quad =$$

$$\{(n > 0 \land m \geq 0) \to (r = n^0 \land 0 \leq 0 \leq m \land n > 0)[r \mapsto 1]\} \quad =$$

$$\{(n > 0 \land m \geq 0) \to (1 = n^0 \land 0 \leq 0 \leq m \land n > 0)\}$$

Running VC

We now have a verification condition already generated.

$$\{(n > 0 \land m \geq 0) \rightarrow (1 = n^0 \land 0 \leq 0 \leq m \land n > 0)\} \tag{1}$$

$$\cup$$

$$\mathbf{VC}(\{r = n^0 \land 0 \leq 0 \leq m \land n > 0\}\ i := 0;\ \mathit{WHILE}\ \{r = n^m\}) \tag{2}$$

We need to process (2), that is, the second call to **VC**:

$$\mathbf{VC}(\{r = n^0 \land 0 \leq 0 \leq m \land n > 0\}\ i := 0;\ \mathit{WHILE}\ \{r = n^m\}) =$$

$$\mathbf{VC}(\{r = n^0 \land 0 \leq 0 \leq m \land n > 0\}\ i := 0\ \{\mathit{wprec}(\mathit{WHILE}, r = n^m)\}) \tag{3}$$

$$\cup$$

$$\mathbf{VC}(\{\mathbf{wprec}(\mathit{WHILE}, r = n^m)\}\ \mathit{WHILE}\ \{r = n^m\}) \tag{4}$$

In the next slides we will address (3) and (4).

Running VC

Starting with (3) we have:

$$\textbf{VC}(\{r = n^0 \wedge 0 \leq 0 \leq m \wedge n > 0\}\; i := 0\; \{\textbf{wprec}(\mathit{WHILE}, r = n^m)\}) \quad =$$

$$\{(r = n^0 \wedge 0 \leq 0 \leq m \wedge n > 0) \to (\textbf{wprec}(\mathit{WHILE}, r = n^m))[i \mapsto 0]\} \quad =$$

$$\{(r = n^0 \wedge 0 \leq 0 \leq m \wedge n > 0) \to (r = n^i \wedge 0 \leq i \leq m \wedge n > 0)[i \mapsto 0]\} \quad =$$

$$\{(r = n^0 \wedge 0 \leq 0 \leq m \wedge n > 0) \to (r = n^0 \wedge 0 \leq 0 \leq m \wedge n > 0)\}$$

We are now done with another VC. Next we proceed to solve (4):

$$\textbf{VC}(\{\textbf{wprec}(\mathit{WHILE}, r = n^m)\}\; \mathit{WHILE}\; \{r = n^m\})$$

Running VC

Now, picking up on (4) we have:

$$\mathbf{VC}(\{\mathbf{wprec}(\mathit{WHILE}, r = n^m)\}\ \mathit{WHILE}\ \{r = n^m\}) =$$

$$\mathbf{VC}(\{r = n^i \wedge 0 \leq i \leq m \wedge n > 0\}\ \mathit{WHILE}\ \{r = n^m\}) =$$

$$\{(r = n^i \wedge 0 \leq i \leq m \wedge n > 0) \rightarrow (r = n^i \wedge 0 \leq i \leq m \wedge n > 0)\}$$
$$\cup$$
$$\{(r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i \leq m)) \rightarrow (r = n^m)\}$$
$$\cup$$
$$\mathbf{VC}(\{r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i < m)\}\ r := r * n;\ i := i + 1\ \{r = n^m\}) \quad (5)$$

We now have to process (5) since the other parts are already generated VCs.

Running VC

Now, picking up on (5) we have:

$$\mathbf{VC}(\{(r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i < m))\}\, r := r * n;\ i := i + 1\, \{r = n^m\}) \qquad =$$

$$\mathbf{VC}(\{r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i < m)\}\, r := r * n\, \{\mathbf{wprec}(i := i + 1, r = n^m)\})$$
$$\cup$$
$$\mathbf{VC}(\{\mathbf{wprec}(i := i + 1, r = n^m)\}\, i := i + 1\, \{r = n^m\}) \qquad =$$

$$\mathbf{VC}(\{r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i \leq m)\}\, r := r * n\, \{(r = n^m)[i \mapsto i + 1]\})$$
$$\cup$$
$$\mathbf{VC}(\{(r = n^m)[i \mapsto i + 1]\}\, i := i + 1\, \{r = n^m\})$$

Running VC

Continuing with what is left:

$$\begin{array}{c} \mathbf{VC}(\{(r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i < m))\}\ r := r * n\ \{(r = n^m)\}) \\ \cup \\ \mathbf{VC}(\{(r = n^m)\}\ i := i + 1\ \{r = n^m\}) \end{array} \quad =$$

$$\begin{array}{c} \{(r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i < m)) \rightarrow (r = n^m)[r \rightarrow r * n]\} \\ \cup \\ \{(r = n^m) \rightarrow (r = n^m)[i \mapsto i + 1]\} \end{array} \quad =$$

$$\begin{array}{c} \{(r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i < m)) \rightarrow (r * n = n^m)\} \\ \cup \\ \{(r = n^m) \rightarrow (r = n^m)\} \end{array}$$

We now have processed all the VCs.

Running VC

Continuing with what is left:

$$
\begin{aligned}
\{ \\
&(n > 0 \wedge m \geq 0) \rightarrow (1 = n^0 \wedge 0 \leq 0 \leq m \wedge n > 0)\ , \\
&(r = n^0 \wedge 0 \leq 0 \leq m \wedge n > 0) \rightarrow (r = n^0 \wedge 0 \leq 0 \leq m \wedge n > 0)\ , \\
&(r = n^i \wedge 0 \leq i \leq m \wedge n > 0) \rightarrow (r = n^i \wedge 0 \leq i \leq m \wedge n > 0)\ , \\
&(r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i < m)) \rightarrow (r = n^m)\ , \\
&(r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i < m)) \rightarrow (r * n = n^m)\ , \\
&(r = n^m) \rightarrow (r = n^m) \qquad \}
\end{aligned}
$$

Final result

With this set of VCs the next step is to understand if they are all valid. One can see immediately that the ones in blue are trivially valid

$$
\begin{aligned}
\{ \\
&(n > 0 \land m \geq 0) \rightarrow (1 = n^0 \land 0 \leq 0 \leq m \land n > 0)\ , \\
&(r = n^0 \land 0 \leq 0 \leq m \land n > 0) \rightarrow (r = n^0 \land 0 \leq 0 \leq m \land n > 0)\ , \\
&(r = n^i \land 0 \leq i \leq m \land n > 0) \rightarrow (r = n^i \land 0 \leq i \leq m \land n > 0)\ , \\
&(r = n^i \land 0 \leq i \leq m \land n > 0 \land \neg(i \leq m)) \rightarrow (r = n^m)\ , \\
&(r = n^i \land 0 \leq i \leq m \land n > 0 \land \neg(i \leq m)) \rightarrow (r * n = n^m)\ , \\
&(r = n^m) \rightarrow (r = n^m) \qquad \}
\end{aligned}
$$

Final result

For the remaining two VCs, we need to reason a bit. Let us first look into

$$(r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i \leq m)) \rightarrow (r = n^m)$$

which is the same as

$$(r = n^i \wedge 0 \leq i \wedge i \leq m \wedge n > 0 \wedge i > m) \rightarrow (r = n^m)$$

But we have a contradiction of the hypoteses, here marked in red:

$$(r = n^i \wedge 0 \leq i \wedge \color{red}{i \leq m} \wedge n > 0 \wedge \color{red}{i > m}) \rightarrow (r = n^m)$$

Hence, we can derive for this inconsistency that

$$\textbf{false} \rightarrow r = n^m$$

Which trivially holds. The same reason is applicable to the remaining VC, which leads us to conclude that the set of VCs generated is valid and thus our program is correct wrt. the prescribed Hoare triple!!!

Using the improved VC algorithm

We will now apply the VC algorithm to generate all proof obligations.

$$
\begin{aligned}
&\textbf{VCG}(\{n > 0 \land m \geq 0\}\; r := 1; i := 0;\; \mathit{WHILE}, \{r = n^m\}) \\
&= \\
&\{(n > 0 \land m \geq 0) \to \textbf{wprec}(r := 1; i := 0;\; \mathit{WHILE}, r = n^m)\} \quad (1) \\
&\cup \\
&\textbf{VC}(r := 1; i := 0;\; \mathit{WHILE}, r = n^m) \quad (2)
\end{aligned}
$$

As done before, lets proceed with (1) first and then continue with (2).

Using the improved VC algorithm

The case of (1) amounts at processing the wprec function.

$$
\begin{array}{rr}
\{(n > 0 \land m \geq 0) \rightarrow \mathbf{wprec}(r := 1; i := 0;\ WHILE, r = n^m)\} & = \\
\{(n > 0 \land m \geq 0) \rightarrow \mathbf{wprec}(r := 1, \mathbf{wprec}(i := 0, \mathbf{wprec}(WHILE, r = n^m)))\} & = \\
\{(n > 0 \land m \geq 0) \rightarrow \mathbf{wprec}(r := 1, \mathbf{wprec}(i := 0, r = n^i \land 0 \leq i \leq m \land n > 0))\} & = \\
\{(n > 0 \land m \geq 0) \rightarrow \mathbf{wprec}(r := 1, (r = n^i \land 0 \leq i \leq m \land n > 0)[i \mapsto 0])\} & = \\
\{(n > 0 \land m \geq 0) \rightarrow ((r = n^i \land 0 \leq i \leq m \land n > 0)[i \mapsto 0, r \mapsto 1]\} & = \\
\{(n > 0 \land m \geq 0) \rightarrow (1 = n^0 \land 0 \leq 0 \leq m \land n > 0)\} &
\end{array}
$$

Now we continue with (2) since we have finished obtaining a VC from (1).

Using the improved VC algorithm

Now, let us continue with the expansion of (2):

$$
\begin{aligned}
\mathbf{VC}(r := 1; i := 0;\ \mathit{WHILE}, r = n^m) & \quad = \\
\mathbf{VC}(r := 1, \mathbf{wprec}(i := 0;\ \mathit{WHILE}, r = n^m)) & \\
\cup & \\
\mathbf{VC}(i := 0;\ \mathit{WHILE}, r = n^m) & \quad = \\
\emptyset \cup \mathbf{VC}(i := 0;\ \mathit{WHILE}, r = n^m) & \quad = \\
\mathbf{VC}(i := 0, \mathbf{wprec}(\mathit{WHILE}, r = n^m)) & \quad (3) \\
\cup & \\
\mathbf{VC}(\mathit{WHILE}, r = n^m) & \quad (4)
\end{aligned}
$$

Now we continue with (3) and (4).

Using the improved VC algorithm

Picking up first on (3), we have:

$$\mathbf{VC}(r := 1, \mathbf{wprec}(i := 0;\ \mathit{WHILE}, r = n^m)) \quad = \quad \emptyset$$

Next, we have to deal with (4):

$$\begin{gathered}\mathbf{VC}(\mathit{WHILE}, r = n^m) \\ = \\ \{r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge i < m \rightarrow \mathbf{wprec}(r := r * n; i := i + 1, r = n^i \wedge 0 \leq i \leq m \wedge n > 0)\} \\ \cup \\ \mathbf{VC}(r := r * n; i := i + 1, r = n^i \wedge 0 \leq i \leq m \wedge n > 0)\ (5) \\ \cup \\ \{(r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge \neg(i < m)) \rightarrow r = n^m\}\end{gathered}$$

Now let us expand (5)

Using the improved VC algorithm

Expanding (5), we have:

$$\mathbf{VC}(r := r * n; i := i + 1, r = n^i \wedge 0 \leq i \leq m \wedge n > 0) \qquad =$$

$$\mathbf{VC}(r := r * n, \mathbf{wprec}(i := i + 1, r = n^i \wedge 0 \leq i \leq m \wedge n > 0))$$
$$\cup$$
$$\mathbf{VC}(i := i + 1, r = n^i \wedge 0 \leq i \leq m \wedge n > 0) \qquad =$$

$$\emptyset \cup \emptyset = \emptyset$$

We are done with (5). Now we have to finish the VC still containing call to wprec:

$$\{r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge i < m \rightarrow \mathbf{wprec}(r := r * n; i := i + 1, r = n^i \wedge 0 \leq i \leq m \wedge n > 0)\}$$

Using the improved VC algorithm

Let us continue with what is missing:

$$\{r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge i < m \rightarrow \mathbf{wprec}(r := r * n; i := i + 1, r = n^i \wedge 0 \leq i \leq m \wedge n > 0)\}$$

$$= \text{(applying twice the definition of } \mathbf{wprec}\text{)}$$

$$\{r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge i < m \rightarrow (r = n^i \wedge 0 \leq i \leq m \wedge n > 0)[i \mapsto i + 1][r \mapsto r * n]\}$$
$$=$$
$$\{r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge i < m \rightarrow (r = n^{i+1} \wedge 0 \leq i \leq m \wedge n > 0)[r \mapsto r * n]\}$$
$$=$$
$$\{r = n^i \wedge 0 \leq i \leq m \wedge n > 0 \wedge i < m \rightarrow (r * n = n^{i+1} \wedge 0 \leq i \leq m \wedge n > 0)\}$$

Using the improved VC algorithm

Now we can join all VCs together and check if this set is valid (and thus the code is correct wrt to the given specification/Hoare triple):

$$\mathbf{VC}(\{n > 0 \land m \geq 0\}\; r := 1; i := 0;\; \mathit{WHILE}, \{r = n^m\}) \quad =$$

$$\{(n > 0 \land m \geq 0) \rightarrow (1 = n^0 \land 0 \leq 0 \leq m \land n > 0)\} \tag{1}$$

$$\cup$$

$$\{(r = n^i \land 0 \leq i \leq m \land n > 0 \land \neg(i < m)) \rightarrow r = n^m\} \tag{2}$$

$$\cup$$

$$\{r = n^i \land 0 \leq i \leq m \land n > 0 \land i < m \rightarrow (r * n = n^{i+1} \land 0 \leq i \leq m \land n > 0)\} \quad (2)$$

Final result of using the improved VC algorithm

The first thing to notice is that indeed the improved version of VC generates less verification conditions. Regarding their validity:

- (1) is trivially true, since $1 = n^0$ and the remaining parts of the conjunction are part of the hypotheses;
- the first formula marked as (2) is also trivially correct due to the contradiction on the hypothesis, that is, $\neg(i < m)$ amounts at $i \geq m$ but $i \leq m$ in the hypothesis as well.
- for the second formula marked as (2), we can easily show that $r * n = n^{i+1} \leftrightarrow r * n = n^i * n$ and thus we can conclude from this that $r = n^i$, which is part of the assumptions, hence it is trivially true.