

Appendix: Formal proof of completeness of inlining

Abstract

We prove the completeness theorem in this appendix. We reuse the definitions and lemmas from the Isabelle/HOL formalization of the soundness proof. Section 2 formalizes the bounded semantics. Section 3 formalizes the maximally annotated property. Section 4 formally expresses the theorem, and the proof strategy: We define a completeness invariant, which implies the completeness theorem, and we prove this invariant by induction. Finally, each section afterwards proves a case of the induction proof.

Contents

1	Differences in notations with the paper	1
2	Bounded semantics	1
3	Maximally annotated property	2
3.1	Methods calls in the inlined programs are unambiguously indexed	2
3.2	Projection	2
3.3	Collecting method annotations	3
3.4	Maximally annotated: Formal definition	3
4	Completeness	4
4.1	Theorem	4
4.2	Completeness invariant (induction proof)	5
5	Induction case: Loop-free and method calls-free statements	5
6	Induction case: Non-deterministic branching	5
6.1	Point 1	6
6.2	Point 2	6
6.3	Point 3	6
7	Induction case: Sequential composition	6
7.1	Point 1	7
7.2	Point 2	7
7.3	Point 3	7
8	Induction case: Method ($depth = n = 0$)	7
9	Induction case: Loop ($depth = n = 0$)	8
10	Induction case: Method ($depth = n + 1$)	8
10.1	Proving point 1 and 2	9
10.2	Proving point 3	10
11	Induction case: Loop ($depth = n + 1$)	10
11.1	Proving point 4	11
11.2	Dividing point 1	11
11.3	Proving point 1.1	11
11.4	Dividing point 2	12
11.5	Dividing point 3	12
11.6	Proving points 2.1 and 3.1	13
11.7	First case: $A' \neq \emptyset$	13
11.7.1	Proving point 1.2	13
11.7.2	Proving point 2.2 and 3.2	13
11.8	Second case: $A' = \emptyset$	13
11.8.1	Proving point 3.2	14

1 Differences in notations with the paper

This appendix uses the notations from Isabelle to express the semantics, which are slightly different than the ones from the paper, which were used for simplicity. In this appendix, the function sem takes as input a set of states and a statement, and returns a set of states

$$sem_M : \mathcal{P}(\Sigma) \times Stmt \rightarrow \mathcal{P}(\Sigma)$$

Furthermore, $sem_M(A, s)$ is undefined if there exists a state in A for which s does not verify.

Similarly, the function ver takes as input a set of states and a statement, and returns a boolean:

$$ver_M : \mathcal{P}(\Sigma) \times Stmt \rightarrow Boolean$$

$ver_M(A, s)$ holds if and only if the statement s verifies in all states of A .

2 Bounded semantics

As explained in the paper, we need a *bounded semantics* that stops executions when the parameter n of the inlining reaches 0. The states φ that we consider for the execution all define a special variable dep , which we also represent as a property of a state given by the following function:

$$dep : \Sigma \mapsto \mathbb{N}$$

This special variable dep cannot be modified by most statements. It is only modified by the bounded semantics itself, when dealing with loops and method calls.

The bounded semantics is defined with the two functions $semB_M : \mathcal{P}(\Sigma) \times Stmt \rightarrow \mathbb{P}(\Sigma)$ and $verB_M : \mathcal{P}(\Sigma) \times Stmt \rightarrow Bool$. For a set of states A and a statement s , $verB_M(A, s)$ holds if and only if s verifies with all states in A in the bounded semantics. In this case, $semB_M(A, s)$ is the set of states resulting from the execution of s in the states in A with the bounded semantics. The definitions of $verB$ and $semB$ are similar to the ones of ver and sem for most cases, except for method calls and loops.

The notation **assign** b (where b is an assertion) is syntatic sugar for **havoc** dep ; **assume** b . The bounded semantics of method calls is:

$$verB_M(\{\varphi\}, \vec{y} := m(\vec{x})) = \begin{cases} \top & \text{if } dep(\varphi) = 0 \\ ver_M(\{\varphi\}, dep -= 1 ; \vec{y} := m(\vec{x}) ; dep += 1) & \text{otherwise} \end{cases}$$

$$semB_M(\{\varphi\}, \vec{y} := m(\vec{x})) = \begin{cases} \emptyset & \text{if } dep(\varphi) = 0 \\ sem_M(\{\varphi\}, dep -= 1 ; \vec{y} := m(\vec{x}) ; dep += 1) & \text{otherwise} \end{cases}$$

and for loops (where $\vec{l} := modif(s) \cap \sigma(\varphi)$):

$$verB_M(\{\varphi\}, \text{while } (b) \text{ inv } I \{s\})$$

$$= verB_M(\{|\varphi|\}, \text{havoc } \vec{l} ; \text{assign } 1 \leq dep \leq dep(\varphi) ; \text{inhale } I ; \text{assume } b ; dep -= 1 ; s ; \text{exhale } I)$$

$$\wedge ver_M(\{\varphi\}, \text{exhale } I ; \text{havoc } \vec{l} ; \text{assign } 0 \leq dep \leq dep(\varphi) ; \text{inhale } I ; \text{assume } \neg b ; dep \leftarrow dep(\varphi))$$

and

$$semB_M(\{\varphi\}, \text{while } (b) \text{ inv } I \{s\})$$

$$= sem_M(\{\varphi\}, \text{exhale } I ; \text{havoc } \vec{l} ; \text{assign } 0 \leq dep \leq dep(\varphi) ; \text{inhale } I ; \text{assume } \neg b ; dep \leftarrow dep(\varphi))$$

We also define the following function

$$deps : \mathbb{N} \mapsto \mathcal{P}(\Sigma)$$

where $deps(n)$ is the singleton containing the unique state φ that has no impure resource and defines only the special variable dep with the value n . Thus, $dep(\varphi) = n$ and $\sigma(\varphi) = \{dep\}$.

3 Maximally annotated property

The maximally annotated property holds for an annotated program (s, M) if and only if its annotation is a maximal annotation (w.r.t. its inlined version with a given inlining bound).

3.1 Methods calls in the inlined programs are unambiguously indexed

We express the fact that it is possible to distinguish different states using some local variables with the following definition:

Definition 1. A set of states Ind is mutually disjoint if and only if

$$\forall i, j \in Ind. i \neq j \implies \neg(i \# j)$$

We then define the concept of a partial annotation for a method:

Definition 2. A partial annotation for a method m is a list of quadruples

$$[[m, i_1, P_1, Q_1], \dots, [m, i_k, P_k, Q_k]]$$

with $i_j \in \Sigma$ (indices, pure states), and P_j and Q_j annotations. This partial annotation represents the following annotation:

method $m(\dots)$ **returns** (\dots)

requires $i1 \parallel \dots \parallel ik$

requires $i1 \implies P1$

...

requires $ik \implies Pk$

ensures $i1 \parallel \dots \parallel ik$

ensures $i1 \implies Q1$

...

ensures $ik \implies Qk$

A method with such an annotation verifies if and only if it verifies for all quadruples:

Lemma 1. If the indices of the partial annotation are mutually disjoint, then the method m partially annotated verifies if and only

$$\forall j. 1 \leq j \leq k \implies \{i_j \wedge P_j\} m \{i_j \wedge Q_j\}$$

We also have the property that, if the indices of the assertion are mutually disjoint, then inhaling or exhaling such an assertion in a state which is compatible with one index is the same as inhaling or exhaling the assertion corresponding to the relevant quadruple:

Lemma 2. If the indices of the partial annotation P, Q are mutually disjoint, and $\varphi \# i_j$, then

$$\begin{aligned} ver_M(\{\varphi\}, \text{inhale } Q) &= ver_M(\{\varphi\}, \text{inhale } Q_j) \\ sem_M(\{\varphi\}, \text{inhale } Q) &= sem_M(\{\varphi\}, \text{inhale } Q_j) \\ ver_M(\{\varphi\}, \text{exhale } P) &= ver_M(\{\varphi\}, \text{exhale } P_j) \\ sem_M(\{\varphi\}, \text{exhale } P) &= sem_M(\{\varphi\}, \text{exhale } P_j) \end{aligned}$$

3.2 Projection

We now formally define the concept of projecting a state onto a set of local variables, which is needed to express the maximally annotated property.

Definition 3. φ is a state, V a set of variables. $\pi_{\vec{V}}(\varphi)$ is the state with the same permissions, but where we removed all variables outside of \vec{V} .

$$\pi_{\vec{V}}(A) = \{\pi_{\vec{V}}(\varphi) \mid \varphi \in A\}$$

We then express two simple lemmas, which can be proved from the properties we assume about the verification language and the separation algebra. If a statement does not read any of the variables in \vec{V} , then these variables do not influence the execution of this statement:

Lemma 3. If $read(s) \cap \vec{V} = \emptyset$, then, for all $\varphi \in \Sigma$:

$$\begin{aligned} ver_M(\{\varphi\}, s) &\iff ver_M(\{\pi_{\vec{V}}(\varphi)\}) \\ sem_M(\{\varphi\}, s) &= sem_M(\{\pi_{\vec{V}}(\varphi)\}, s) \oplus \{\pi_{\vec{V}}(|\varphi|)\} \end{aligned}$$

Lemma 4. If $read(s) \subseteq \vec{V}$, then, for all $\varphi \in \Sigma$:

$$\begin{aligned} ver_M(\{\varphi\}, s) &\iff ver_M(\{\pi_{\vec{V}}(\varphi)\}) \\ \pi_{\vec{V}}(sem_M(\{\varphi\}, s)) &= sem_M(\{\pi_{\vec{V}}(\varphi)\}, s) \end{aligned}$$

3.3 Collecting method annotations

To prove that the original annotated program verifies, we need to prove that all methods verify w.r.t. their annotations. The completeness invariant, which we define below, states that the methods of M verify w.r.t. to the partial maximal annotation, corresponding to the program. We define here the *collectAn* function, which constructs the partial annotation corresponding to a program and its inlined version. It is parametrized by a set of states A , which corresponds to the states in which the inlined version of the program is executed.

The variables of this function are as follows:

- M is the set of methods of the program.
- n is the depth up to which we inline.
- \vec{U} is the set of variable names already used in the inlining. It is used for avoiding the capture of variables when inlining a method call and renaming its body.

- \vec{V} is the set of variable names readable in the current method (since a method body does not have access to variables not passed via the arguments).
- A is the current set of states in the execution of the inlined program.
- s is the statement we inline.

Definition 4. *collectAn*

- *Method call* ($depth = n + 1$)
 $collectAn_M^{n+1}(\vec{U}, \vec{V}, A, \vec{y} := m(\vec{x})) := \{(m, ind, P_{ind}, Q_{ind})\} \cup collectAn_M^n(\vec{U} \cup \text{modif}(s'), \text{read}(s'), A, s')$
where
 - m corresponds to $(\overrightarrow{args}, \overrightarrow{rets}, _, _, s)$ in the set of methods M
 - $t := (\overrightarrow{args} \cup \overrightarrow{rets}, x \cup y, \vec{U}, \text{modif}(s))$
 - $s' := \text{rename}_t(s)$
 - $A' := \text{sem}_M(A, \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s'))$
 - $\text{Inh}(P_{ind}) = \pi_{\overrightarrow{args} \cup \overrightarrow{rets}}(\text{rename}_{t-1}(A)) \oplus \text{deps}(n)$
 - $\text{Inh}(Q_{ind}) = \pi_{\overrightarrow{args} \cup \overrightarrow{rets}}(\text{rename}_{t-1}(A')) \oplus \text{deps}(n)$
- *Loop* ($depth = n + 1$)
 $collectAn_M^{n+1}(\vec{U}, \vec{V}, A, \text{while } (b) \text{ inv } I \{s\})$
 $:= collectAn_M^n(\vec{U}, \vec{V}, f_b(A), s) \cup collectAn_M^n(\vec{U} \cup \text{modif}(s'), \vec{V}, \text{sem}_M(f_b(A), s'), \text{while } (b) \text{ inv } I \{s\})$
where $s' := \text{inl}_M^n(\vec{U}, s)$ and $A' := \text{sem}_M(f_b(A), s')$
- *Sequential composition*
 $collectAn_M^n(\vec{U}, \vec{V}, A, s_1 ; s_2) := collectAn_M^n(\vec{U}, \vec{V}, A, s_1) \cup collectAn_M^n(\vec{U} \cup \text{read}(s'_1), \vec{V}, \text{sem}_M(A, s'_1), s_2)$
where $s'_1 := \text{inl}_M^n(\vec{U}, s_1)$
- *Non-deterministic branching*
 $collectAn_M^n(\vec{U}, \vec{V}, A, \text{if } (*) \{s_1\} \text{ else } \{s_2\}) := collectAn_M^n(\vec{U}, \vec{V}, A, s_1) \cup collectAn_M^n(\vec{U}, \vec{V}, A, s_2)$
- *Other cases*
 $collectAn_M^n(\vec{U}, \vec{V}, A, s) := \emptyset$ *otherwise*

The most interesting case, the method call case, states that the partial annotation corresponding to this method call is the union of the quadruple $(m, ind, P_{ind}, Q_{ind})$ with the partial annotation corresponding to the inlining of the body of this method. P_{ind} is the assertion which captures the set of states from the execution before the method call, but renamed and projected onto the arguments of the method m . Similarly, Q_{ind} is the assertion which captures the set of states from the execution after the method call, but renamed and for and projected onto the arguments of the method m .

3.4 Maximally annotated: Formal definition

We finally define the maximally annotated property. This property expresses that loop invariants capture the sets of states (up to the variables \vec{U}) from the execution before and after each inlined loop iteration with the right indexing, and method preconditions and postconditions capture the sets of states (up to the variables \vec{U}) from the execution before and after the inlined method calls, also with the right indexing.

Definition 5. *maxAnnot*

- *Method call* ($depth = n + 1$)
 $maxAnnot_M^{n+1}(\vec{U}, \vec{V}, A, \vec{y} := m(\vec{x}))$
 $\iff (\exists (m, ind, P_{ind}, Q_{ind}) \in collectAn_M^{n+1}(\vec{U}, \vec{V}, A, \vec{y} := m(\vec{x})). A \# |ind| \wedge A' \# |ind| \wedge \sigma(|ind|) \subseteq \vec{x}$
 $\wedge \text{Inh}(P_{ind}) = \pi_{\overrightarrow{args} \cup \overrightarrow{rets}}(\text{rename}_{t-1}(A)) \oplus \text{deps}(n) \wedge \text{Inh}(Q_{ind}) = \pi_{\overrightarrow{args} \cup \overrightarrow{rets}}(\text{rename}_{t-1}(A')) \oplus \text{deps}(n))$
 $\wedge \text{rename}_{t-1}(|ind|) \oplus \text{Inh}(P) = \text{rename}_{t-1}(|ind|) \oplus \text{Inh}(P_{ind})$
 $\wedge \text{rename}_{t-1}(|ind|) \oplus \text{Inh}(Q) = \text{rename}_{t-1}(|ind|) \oplus \text{Inh}(Q_{ind}) \wedge maxAnnot_M^n(\vec{U} \cup \text{modif}(s'), \text{read}(s'), A, s')$
where
 - $t := (\overrightarrow{args} \cup \overrightarrow{rets}, x \cup y, \vec{U}, \text{modif}(s))$
 - $s' := \text{rename}_t(s)$
 - $A' := \text{sem}_M(A, \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s'))$
- *Loop* ($depth = n + 1$)
 $maxAnnot_M^{n+1}(\vec{U}, \vec{V}, A, \text{while } (b) \text{ inv } I \{s\})$
 $\iff (\exists ind \in \Sigma. A \# |ind| \wedge |ind| \oplus \text{deps}(n + 1) \oplus \text{Inh}(I) = \pi_V(A) \wedge \sigma(ind) \cap \text{modif}(s) = \emptyset$
 $\wedge (\exists ind \in \Sigma. A' \# |ind| \wedge |ind| \oplus \text{deps}(n + 1) \oplus \text{Inh}(I) = \pi_V(A') \wedge \sigma(ind) \cap \text{modif}(s) = \emptyset)$
 $\wedge maxAnnot_M^n(\vec{U}, \vec{V}, f_b(A), s) \wedge maxAnnot_M^n(\vec{U} \cup \text{modif}(s'), \vec{V}, \text{sem}_M(f_b(A), s'), \text{while } (b) \text{ inv } I \{s\})$
where
 - $s' := \text{inl}_M^n(\vec{U}, s)$

- $A' := \text{sem}_M(f_b(A), s')$
- *Loop* ($\text{depth} = 0$)
$$\text{maxAnnot}_M^0(\vec{U}, \vec{V}, A, \mathbf{while} (b) \mathbf{inv} I \{s\}) \iff (\exists \text{ind} \in \Sigma.A \# |\text{ind}| \wedge |\text{ind}| \oplus \text{deps}(0) \oplus \text{Inh}(I) = \pi_V(A) \wedge \sigma(\text{ind}) \cap \text{modif}(s) = \emptyset)$$
- *Sequential composition*

$$\begin{aligned} & \text{maxAnnot}_M^n(\vec{U}, \vec{V}, A, s_1 ; s_2) \\ & \iff \text{maxAnnot}_M^n(\vec{U}, \vec{V}, A, s_1) \wedge \text{maxAnnot}_M^n(\vec{U} \cup \text{read}(s'_1), \vec{V}, \text{sem}_M(A, s'_1), s_2) \quad (\text{where } s'_1 := \text{inl}_M^n(\vec{U}, s_1)) \end{aligned}$$
- *Non-deterministic branching*

$$\text{maxAnnot}_M^n(\vec{U}, \vec{V}, A, \mathbf{if} (*) \{s_1\} \mathbf{else} \{s_2\}) \iff \text{maxAnnot}_M^n(\vec{U}, \vec{V}, A, s_1) \wedge \text{maxAnnot}_M^n(\vec{U}, \vec{V}, A, s_2)$$
- *Other cases*

$$\text{maxAnnot}_M^n(\vec{U}, \vec{V}, A, s) \iff \top$$

4 Completeness

4.1 Theorem

We now express the completeness theorem, where $\text{verMethodsB}(M)$ expresses that all methods of M verify w.r.t. their annotations, and $\text{annot}(M, \text{collectAn}_M^n(\text{modif}(s), \text{read}(s), \{u\}, s))$ is the set of methods M that have been annotated with the quadruples $\text{collectAn}_M^n(\text{modif}(s), \text{read}(s), \{u\}, s)$. The annotation corresponding to $\text{collectAn}_M^n(\text{modif}(s), \text{read}(s), \{u\}, s)$ is the same as the one expressed in the maxAnnot function.

Theorem 1. *If*

1. *The program is well-formed:* $\text{wfStmnt}_M(s) \wedge \text{wfMethods}(M)$
2. *The program is maximally annotated:* $\text{maxAnnot}_M^n(\text{modif}(s), \text{read}(s), \{u\}, s)$
3. *Loop iterations do not create new variables.*
4. $\text{ver}_M(\{u\}, \text{inl}_M^n(\text{modif}(s), s))$

Then

1. $\text{verB}_M(\{\text{deps}(n)\}, s)$
2. $\text{verMethodsB}(\text{annot}(M, \text{collectAn}_M^n(\text{modif}(s), \text{read}(s), \{u\}, s)))$

4.2 Completeness invariant (induction proof)

To prove the completeness theorem, we prove the following stronger property by computational induction on the structure of the inline function:

Definition 6. *The completeness invariant, written $\text{CompletenessInv}_M^n(s)$, holds if and only if:*

For all set of states A , sets of variable names \vec{V} and \vec{U} , if

1. $\text{wfStmnt}_M(s) \wedge \text{wfMethods}(M)$
2. $\text{read}(s) \subseteq \vec{V} \wedge \text{dep} \notin \vec{V}$
3. $\text{modif}(s) \subseteq \vec{U} \wedge \text{dom}(A) \subseteq \vec{U}$
4. $\text{maxAnnot}_M^n(\vec{U}, \vec{V}, A, s)$
5. $\text{ver}_M(A, \text{inl}_M^n(\vec{U}, s))$
6. *Loop iterations do not create new variables*

Then

1. $\text{verB}_M(A \oplus \text{deps}(n), s)$
2. $\pi_{\vec{V}}(\text{semB}_M(A \oplus \text{deps}(n), s)) = \pi_{\vec{V}}(\text{sem}_M(A, \text{inl}_M^n(\vec{U}, s)))$
3. $\text{verMethodsB}(\text{annot}(M, \text{collectAn}_M^n(\vec{U}, \vec{V}, A, s)))$

By instantiating this invariant with $\vec{V} = \text{read}(s)$, $\vec{U} = \text{modif}(s)$, and $A = \{u\}$, we get Theorem 1 from properties 1 and 3.

5 Induction case: Loop-free and method calls-free statements

This property is trivial for statements which do not contain any loops or any method calls, since the inlined program is the same as the original program, and nothing has to be proved for $\text{collectAn}_M^n(\vec{U}, \vec{V}, A, s)$ since it is empty.

6 Induction case: Non-deterministic branching

Lemma 5. *If*

1. $CompletenessInv_M^n(s_1)$
2. $CompletenessInv_M^n(s_2)$

then

$$CompletenessInv_M^n(\text{if } (*) \{s_1\} \text{ else } \{s_2\})$$

Proof. Let $s := (\text{if } (*) \{s_1\} \text{ else } \{s_2\})$, we assume $CompletenessInv_M^n(s_1)$ and $CompletenessInv_M^n(s_2)$.

Let A be a set of states, \vec{V} and \vec{U} sets of variable names.

To prove the invariant, we assume that

1. $wfStmt_M(s) \wedge wfMethods(M)$
2. $read(s) \subseteq \vec{V} \wedge dep \notin \vec{V}$
3. $modif(s) \subseteq \vec{U} \wedge dom(A) \subseteq \vec{U}$
4. $maxAnnot_M^n(\vec{U}, \vec{V}, A, s)$
5. $ver_M(A, inl_M^n(\vec{U}, s))$
6. Loop iterations do not create new variables

We need to prove the following points:

1. $verB_M(A \oplus deps(n), \text{if } (*) \{s_1\} \text{ else } \{s_2\})$
2. $\pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, \text{if } (*) \{s_1\} \text{ else } \{s_2\}))) = \pi_{\vec{V}}(semB_M(A \oplus deps(n), \text{if } (*) \{s_1\} \text{ else } \{s_2\}))$
3. $verMethodsB(annot(M, collectAn_M^n(\vec{U}, \vec{V}, A, \text{if } (*) \{s_1\} \text{ else } \{s_2\})))$

We have $ver_M(A, inl_M^n(\vec{U}, s_1))$. From $CompletenessInv_M^n(s_1)$, we get

$$verB_M(A \oplus deps(n), s_1) \tag{1}$$

$$\pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, s_1))) = \pi_{\vec{V}}(semB_M(A \oplus deps(n), s_1)) \tag{2}$$

$$verMethodsB(annot(M, collectAn_M^n(\vec{U}, \vec{V}, A, s_1))) \tag{3}$$

We have $ver_M(A, inl_M^n(\vec{U}, s_2))$. From $CompletenessInv_M^n(s_2)$, we get

$$verB_M(A \oplus deps(n), s_2) \tag{4}$$

$$\pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, s_2))) = \pi_{\vec{V}}(semB_M(A \oplus deps(n), s_2)) \tag{5}$$

$$verMethodsB(annot(M, collectAn_M^n(\vec{U}, \vec{V}, A, s_2))) \tag{6}$$

6.1 Point 1

$$verB_M(A, s) \iff verB_M(A, s_1) \wedge verB_M(A, s_2)$$

We conclude using Equation 1 and Equation 4.

6.2 Point 2

$$\begin{aligned} \pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, s))) &= \pi_{\vec{V}}(sem_M(A, \text{if } (*) \{inl_M^n(\vec{U}, s_1)\} \text{ else } \{inl_M^n(\vec{U}, s_2)\})) \\ &= \pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, s_1)) \cup sem_M(A, inl_M^n(\vec{U}, s_2))) \\ &= \pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, s_1))) \cup \pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, s_2))) \\ &= \pi_{\vec{V}}(semB_M(A, s_1)) \cup \pi_{\vec{V}}(semB_M(A, s_2)) \quad (\text{Equations 2 and 5}) \\ &= \pi_{\vec{V}}(semB_M(A, s_1) \cup semB_M(A, s_2)) \\ &= \pi_{\vec{V}}(semB_M(A, s)) \end{aligned}$$

6.3 Point 3

Since $collectAn_M^n(\vec{U}, \vec{V}, A, s) = collectAn_M^n(\vec{U}, \vec{V}, A, s_1) \cup collectAn_M^n(\vec{U}, \vec{V}, A, s_2)$, we have

$$\begin{aligned} &verMethodsB(annot(M, collectAn_M^n(\vec{U}, \vec{V}, A, s))) \\ &\iff verMethodsB(annot(M, collectAn_M^n(\vec{U}, \vec{V}, A, s_1))) \wedge verMethodsB(annot(M, collectAn_M^n(\vec{U}, \vec{V}, A, s_2))) \end{aligned}$$

which corresponds to Equation 3 and Equation 6. □

7 Induction case: Sequential composition

Lemma 6. *If*

1. $CompletenessInv_M^n(s_1)$
2. $CompletenessInv_M^n(s_2)$

then

$$CompletenessInv_M^n(s_1 ; s_2)$$

Proof. Let $s := (s_1 ; s_2)$, we assume $CompletenessInv_M^n(s_1)$ and $CompletenessInv_M^n(s_2)$.

Let A be a set of states, \vec{V} and \vec{U} sets of variable names.

To prove the invariant, we assume that

1. $wfStmt_M(s) \wedge wfMethods(M)$
2. $read(s) \subseteq \vec{V} \wedge dep \notin \vec{V}$
3. $modif(s) \subseteq \vec{U} \wedge dom(A) \subseteq \vec{U}$
4. $maxAnnot_M^n(\vec{U}, \vec{V}, A, s)$
5. $ver_M(A, inl_M^n(\vec{U}, s))$
6. Loop iterations do not create new variables

We need to prove the following points:

1. $verB_M(A \oplus deps(n), s_1 ; s_2)$
2. $\pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, s_1 ; s_2))) = \pi_{\vec{V}}(semB_M(A \oplus deps(n), s_1 ; s_2))$
3. $verMethodsB(annot(M, collectAn_M^n(\vec{U}, \vec{V}, A, s_1 ; s_2)))$

We have $ver_M(A, inl_M^n(\vec{U}, s_1))$. From $CompletenessInv_M^n(s_1)$, we get

$$verB_M(A \oplus deps(n), s_1) \tag{7}$$

$$\pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, s_1))) = \pi_{\vec{V}}(semB_M(A \oplus deps(n), s_1)) \tag{8}$$

$$verMethodsB(annot(M, collectAn_M^n(\vec{U}, \vec{V}, A, s_1))) \tag{9}$$

Let $s'_1 := inl_M^n(\vec{U}, s_1)$, and $A' := sem_M(A, s'_1)$. From $ver_M(A, s'_1 ; inl_M^n(\vec{U} \cup read(s'_1), s_2))$ we get $ver_M(A', inl_M^n(\vec{U}, s_2))$. From $CompletenessInv_M^n(s_2)$, we get

$$verB_M(A' \oplus deps(n), s_2) \tag{10}$$

$$\pi_{\vec{V}}(sem_M(A', inl_M^n(\vec{U} \cup read(s'_1), s_2))) = \pi_{\vec{V}}(semB_M(A' \oplus deps(n), s_2)) \tag{11}$$

$$verMethodsB(annot(M, collectAn_M^n(\vec{U} \cup read(s'_1), \vec{V}, A', s_2))) \tag{12}$$

7.1 Point 1

$$\begin{aligned} verB_M(A, s_1 ; s_2) &\iff verB_M(A, s_1) \wedge verB_M(sem_M(A, s_1), s_2) \\ &\iff verB_M(A, s_1) \wedge verB_M(\pi_{\vec{V}}(sem_M(A, s_1)), s_2) && \text{(Since } read(s_2) \subseteq \vec{V} \text{)} \\ &\iff verB_M(A, s_1) \wedge verB_M(\pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, s_1))), s_2) && \text{(Using Equation 8)} \\ &\iff verB_M(A, s_1) \wedge verB_M(\pi_{\vec{V}}(A'), s_2) \\ &\iff verB_M(A, s_1) \wedge verB_M(A', s_2) && \text{(Since } read(s_2) \subseteq \vec{V} \text{)} \end{aligned}$$

We conclude using Equation 7 and Equation 10.

7.2 Point 2

$$\begin{aligned} \pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, s_1 ; s_2))) &= \pi_{\vec{V}}(sem_M(A, inl_M^n(\vec{U}, s_1) ; inl_M^n(\vec{U} \cup read(s'_1), s_2))) \\ &= \pi_{\vec{V}}(sem_M(sem_M(A, inl_M^n(\vec{U}, s_1)), inl_M^n(\vec{U} \cup read(s'_1), s_2))) \\ &= \pi_{\vec{V}}(sem_M(A', inl_M^n(\vec{U} \cup read(s'_1), s_2))) \\ &= \pi_{\vec{V}}(semB_M(A', s_2)) && \text{(Using Equation 11)} \\ &= \pi_{\vec{V}}(semB_M(\pi_{\vec{V}}(A'), s_2)) && \text{(Since } read(s_2) \subseteq \vec{V} \text{)} \\ &= \pi_{\vec{V}}(semB_M(\pi_{\vec{V}}(semB_M(A, s_1)), s_2)) && \text{(Using Equation 8)} \\ &= \pi_{\vec{V}}(semB_M(semB_M(A, s_1), s_2)) && \text{(Since } read(s_2) \subseteq \vec{V} \text{)} \\ &= \pi_{\vec{V}}(semB_M(A, s_1 ; s_2)) \end{aligned}$$

7.3 Point 3

Since $collectAn_M^n(\vec{U}, \vec{V}, A, s_1 ; s_2) = collectAn_M^n(\vec{U}, \vec{V}, A, s_1) \cup collectAn_M^n(\vec{U} \cup read(s'_1), \vec{V}, A', s_2)$, we have

$$\begin{aligned} & verMethodsB(annot(M, collectAn_M^n(\vec{U}, \vec{V}, A, s_1 ; s_2))) \\ \iff & verMethodsB(annot(M, collectAn_M^n(\vec{U}, \vec{V}, A, s_1))) \wedge verMethodsB(annot(M, collectAn_M^n(\vec{U} \cup read(s'_1), \vec{V}, A, s_2))) \end{aligned}$$

which corresponds to Equation 9 and Equation 12. \square

8 Induction case: Method ($depth = n = 0$)

Lemma 7.

$$CompletenessInv_M^0(\vec{y} := m(\vec{x}))$$

Proof. Let A be a set of states, \vec{V} and \vec{U} sets of variable names.

To prove the invariant, we assume that

1. $wfStmt_M(\vec{y} := m(\vec{x})) \wedge wfMethods(M)$
2. $read(\vec{y} := m(\vec{x})) \subseteq \vec{V} \wedge dep \notin \vec{V}$
3. $modif(\vec{y} := m(\vec{x})) \subseteq \vec{U} \wedge dom(A) \subseteq \vec{U}$
4. $maxAnnot_M^0(\vec{U}, \vec{V}, A, \vec{y} := m(\vec{x}))$
5. $ver_M(A, inl_M^0(\vec{U}, \vec{y} := m(\vec{x})))$
6. Loop iterations do not create new variables

We need to prove the following points:

1. $verB_M(A \oplus deps(0), \vec{y} := m(\vec{x}))$
True by definition.
2. $\pi_{\vec{V}}(sem_M(A, inl_M^0(\vec{U}, \vec{y} := m(\vec{x})))) = \pi_{\vec{V}}(semB_M(A \oplus deps(0), \vec{y} := m(\vec{x})))$
 - $inl_M^0(\vec{U}, \vec{y} := m(\vec{x})) = \text{assume } \perp$
 - $semB_M(A, \vec{y} := m(\vec{x})) = \emptyset$
 - $\pi_{\vec{V}}(sem_M(A, inl_M^0(\vec{U}, \vec{y} := m(\vec{x})))) = \pi_{\vec{V}}(\emptyset) = \pi_{\vec{V}}(semB_M(A \oplus deps(0), \vec{y} := m(\vec{x})))$
3. $verMethodsB(annot(M, collectAn_M^0(\vec{U}, \vec{V}, A, \vec{y} := m(\vec{x}))))$
Directly follows from $collectAn_M^0(\vec{U}, \vec{V}, A, \vec{y} := m(\vec{x})) = \emptyset$

\square

9 Induction case: Loop ($depth = n = 0$)

Lemma 8.

$$CompletenessInv_M^0(\text{while } (b) \text{ inv } I \{s\})$$

Proof. Let A be a set of states, \vec{V} and \vec{U} sets of variable names.

To prove the invariant, we assume that

1. $wfStmt_M(\text{while } (b) \text{ inv } I \{s\}) \wedge wfMethods(M)$
2. $read(\text{while } (b) \text{ inv } I \{s\}) \subseteq \vec{V} \wedge dep \notin \vec{V}$
3. $modif(\text{while } (b) \text{ inv } I \{s\}) \subseteq \vec{U} \wedge dom(A) \subseteq \vec{U}$
4. $maxAnnot_M^0(\vec{U}, \vec{V}, A, \text{while } (b) \text{ inv } I \{s\})$
5. $ver_M(A, inl_M^0(\vec{U}, \text{while } (b) \text{ inv } I \{s\}))$
6. Loop iterations do not create new variables

Let $\varphi \in A$ and $\vec{l} := modif(s) \cap \sigma(\varphi)$. We have

$$\begin{aligned} & sem_M(\{\varphi\} \oplus deps(0), \text{exhale } I ; \text{havoc } \vec{l} ; \text{assign } 0 \leq dep \leq 0 ; \text{inhale } I) \\ = & sem_M(\{\varphi\} \oplus deps(0), \text{havoc } \vec{l} ; \text{assign } 0 \leq dep \leq 0 ; \text{inhale } I) && \text{(Maximally annotated hypothesis)} \\ = & sem_M(\{\vec{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus deps(0), \text{assign } 0 \leq dep \leq 0 ; \text{inhale } I) \\ = & sem_M(\{\vec{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus deps(0), \text{inhale } I) \\ = & \{\vec{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus deps(0) \oplus Inh(I) \\ = & \{\vec{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus deps(0) \oplus \pi_{\vec{V}}(A) && \text{(Maximally annotated hypothesis)} \end{aligned}$$

Thus, since $\vec{l} \subseteq modif(s) \subseteq read(s) \subseteq \vec{V}$,

$$\pi_{\vec{V}}(\{\varphi\}) \subseteq \pi_{\vec{V}}(sem_M(\{\varphi\} \oplus deps(0), \text{exhale } I ; \text{havoc } \vec{l} ; \text{assign } 0 \leq dep \leq 0 ; \text{inhale } I)) \subseteq \pi_{\vec{V}}(A)$$

Therefore

$$\pi_{\vec{V}}(\text{sem}_M(A \oplus \text{deps}(0), \text{exhale } I ; \text{havoc } \vec{t} ; \text{assign } 0 \leq \text{dep} \leq 0 ; \text{inhale } I)) = A \quad (13)$$

We need to prove the following points:

1. $\text{ver}B_M(A, \text{while } (b) \text{ inv } I \{s\})$
 - $\text{ver}B_M(\{\varphi\}, \text{havoc } \vec{t} ; \text{assign } 1 \leq \text{dep} \leq 0 ; \text{inhale } I ; \text{assume } b ; \text{dep} -= 1 ; s ; \text{exhale } I)$
since $\text{sem}B_M(\{\varphi\}, \text{havoc } \vec{t} ; \text{assign } 1 \leq \text{dep} \leq 0) = \emptyset$
 - $\text{ver}_M(\{\varphi\}, \text{exhale } I ; \text{havoc } \vec{t} ; \text{assign } 0 \leq \text{dep} \leq 0 ; \text{inhale } I ; \text{assume } \neg b ; \text{dep} \leftarrow 0)$
using Equation 13
2. $\pi_{\vec{V}}(\text{sem}_M(A, \text{inl}_M^0(\text{while } (b) \text{ inv } I \{s\}))) = \pi_{\vec{V}}(\text{sem}B_M(A, \text{while } (b) \text{ inv } I \{s\}))$

$$\begin{aligned} \pi_{\vec{V}}(\text{sem}_M(A, \text{inl}_M^0(\text{while } (b) \text{ inv } I \{s\}))) &= \pi_{\vec{V}}(\text{sem}_M(A, \text{assume } \neg b)) \\ &= \pi_{\vec{V}}(f_{\neg b}(A)) \\ &= \pi_{\vec{V}}(\text{sem}B_M(A, \text{while } (b) \text{ inv } I \{s\})) \end{aligned} \quad (\text{Using Equation 13})$$
3. $\text{verMethods}B(\text{annot}(M, \text{collectAn}_M^0(\vec{U}, \vec{V}, A, \text{while } (b) \text{ inv } I \{s\})))$
Directly follows from $\text{collectAn}_M^0(\vec{U}, \vec{V}, A, \text{while } (b) \text{ inv } I \{s\}) = \emptyset$

□

10 Induction case: Method ($\text{depth} = n + 1$)

Lemma 9. *If*

1. *The method name m corresponds to the method $(\overrightarrow{\text{args}}, \overrightarrow{\text{rets}}, P, Q, s)$ in the set of methods M .*
2. *$\text{CompletenessInv}_M^n(\text{rename}_{(\overrightarrow{\text{args}} \cup \overrightarrow{\text{rets}}, \vec{x} \cup \vec{y}, \vec{U}, \text{modif}(s))}(s))$*

Then

$$\text{CompletenessInv}_M^{n+1}(\vec{y} := m(\vec{x}))$$

Proof. Let $t := (\overrightarrow{\text{args}} \cup \overrightarrow{\text{rets}}, \vec{x} \cup \vec{y}, \vec{U}, \text{modif}(s))$ be a renaming configuration.

Let $s' := \text{rename}_t(s)$. We have

$$\text{inl}_M^{n+1}(\vec{y} := m(\vec{x})) = \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s')$$

Let us assume $\text{CompletenessInv}_M^n(s')$.

Let A be a set of states, \vec{V} and \vec{U} sets of variable names.

To prove the invariant, we assume that

1. $\text{wfStmt}_M(\vec{y} := m(\vec{x})) \wedge \text{wfMethods}(M)$
2. $\text{read}(s) \subseteq \vec{V} \wedge \text{dep} \notin \vec{V}$
3. $\text{modif}(s) \subseteq \vec{U} \wedge \text{dom}(A) \subseteq \vec{U}$
4. $\text{maxAnnot}_M^n(\vec{U}, \vec{V}, A, \vec{y} := m(\vec{x}))$
5. $\text{ver}_M(A, \text{inl}_M^n(\vec{U}, \vec{y} := m(\vec{x})))$
6. Loop iterations do not create new variables

Let $P' := \text{rename}_{(\overrightarrow{\text{args}} \cup \overrightarrow{\text{rets}}, \vec{x} \cup \vec{y}, \emptyset, \emptyset)}(P)$, and $Q' := \text{rename}_{(\overrightarrow{\text{args}} \cup \overrightarrow{\text{rets}}, \vec{x} \cup \vec{y}, \emptyset, \emptyset)}(Q)$.

Since $\text{read}(P) \subseteq \vec{x}$, we have $P' = \text{rename}_t(P)$. Similarly, since $\text{read}(Q) \subseteq \vec{x} \cup \vec{y}$, we have $Q' = \text{rename}_t(Q)$.

We need to prove the following points:

1. $\text{ver}B_M(A \oplus \text{deps}(n+1), \text{dep} -= 1 ; \text{exhale } P' ; \text{havoc } \vec{y} ; \text{inhale } Q ; \text{dep} += 1)$
2. $\pi_{\vec{V}}(\text{sem}B_M(A \oplus \text{deps}(n+1), \text{dep} -= 1 ; \vec{y} := m(\vec{x}) ; \text{dep} += 1))$
 $= \pi_{\vec{V}}(\text{sem}_M(A, \text{inl}_M^{n+1}(\vec{U}, \vec{y} := m(\vec{x}))))$
3. $\text{verMethods}B(\text{annot}(M, \text{collectAn}_M^{n+1}(\vec{U}, \vec{V}, A, \vec{y} := m(\vec{x}))))$

From $\text{CompletenessInv}_M^n(s')$, we get

$$\text{ver}B_M(A \oplus \text{deps}(n), s') \quad (14)$$

$$\pi_{\text{read}(s')}(\text{sem}B_M(A \oplus \text{deps}(n), s')) = \pi_{\text{read}(s')}(\text{sem}_M(A, \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s'))) \quad (15)$$

$$\text{verMethods}B(\text{annot}(M, \text{collectAn}_M^n(\vec{U} \cup \text{modif}(s'), \text{read}(s'), A, s'))) \quad (16)$$

Let $\vec{ar} := \overrightarrow{\text{args}} \cup \overrightarrow{\text{rets}}$

From hypothesis 4 ($\text{maxAnnot}_M^n(\vec{U}, \vec{V}, A, \vec{y} := m(\vec{x}))$), there exists $\text{ind}, P_{\text{ind}}, Q_{\text{ind}}$ such that

$$A \# |\text{ind}| \tag{17}$$

$$\text{sem}_M(A, \text{inl}_M^n(s')) \# |\text{ind}| \tag{18}$$

$$\sigma(|\text{ind}|) \subseteq \vec{x} \tag{19}$$

$$\text{rename}_{t-1}(|\text{ind}|) \oplus \text{Inh}(P) = \text{rename}_{t-1}(|\text{ind}|) \oplus \text{Inh}(P_{\text{ind}}) \tag{20}$$

$$\text{rename}_{t-1}(|\text{ind}|) \oplus \text{Inh}(Q) = \text{rename}_{t-1}(|\text{ind}|) \oplus \text{Inh}(Q_{\text{ind}}) \tag{21}$$

and

$$\text{Inh}(P_{\text{ind}}) = \pi_{\text{args} \cup \text{rets}}^{\text{args} \cup \text{rets}}(\text{rename}_{t-1}(A)) \oplus \text{deps}(n) \wedge \text{Inh}(Q_{\text{ind}}) = \pi_{\text{args} \cup \text{rets}}^{\text{args} \cup \text{rets}}(\text{rename}_{t-1}(\text{sem}_M(A, \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s')))) \oplus \text{deps}(n)$$

Therefore we have

$$\text{Inh}(\text{rename}_t(P_{\text{ind}})) = \pi_{\vec{x} \cup \vec{y}}^{\vec{x} \cup \vec{y}}(A) \oplus \text{deps}(n) \tag{22}$$

$$\text{Inh}(\text{rename}_t(Q_{\text{ind}})) = \pi_{\vec{x} \cup \vec{y}}^{\vec{x} \cup \vec{y}}(\text{sem}_M(A, \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s')))) \oplus \text{deps}(n) \tag{23}$$

10.1 Proving point 1 and 2

Let $\varphi \in A$. We have

$$\begin{aligned} & \text{sem}B_M(\{\varphi\} \oplus \text{deps}(n+1), \text{dep} \text{ -- } 1; \text{ exhale } P'; \text{ havoc } \vec{y}; \text{ inhale } Q'; \text{ dep} \text{ ++ } 1) \\ &= \text{sem}B_M(\{\varphi\} \oplus \text{deps}(n), \text{ exhale } P'; \text{ havoc } \vec{y}; \text{ inhale } Q'; \text{ dep} \text{ ++ } 1) \\ &= \text{sem}B_M(\{\varphi\} \oplus \text{deps}(n), \text{ exhale } \text{rename}_t(P); \text{ havoc } \vec{y}; \text{ inhale } Q'; \text{ dep} \text{ ++ } 1) \\ &= \text{sem}B_M(\{\varphi\} \oplus \text{deps}(n), \text{ exhale } \text{rename}_t(P_{\text{ind}}); \text{ havoc } \vec{y}; \text{ inhale } Q'; \text{ dep} \text{ ++ } 1) \quad (\text{Using Equations 17 and 19}) \\ &= \text{sem}B_M(\{|\varphi|\} \oplus \text{deps}(n), \text{ havoc } \vec{y}; \text{ inhale } Q'; \text{ dep} \text{ ++ } 1) \quad (\text{Using Equation 22}) \\ &= \text{sem}B_M(\{\bar{h}_{\vec{y}}(|\varphi|)\} \oplus h(y) \oplus \text{deps}(n), \text{ inhale } Q'; \text{ dep} \text{ ++ } 1) \\ &= \text{sem}B_M(\{\bar{h}_{\vec{y}}(|\varphi|)\} \oplus h(y) \oplus \text{deps}(n), \text{ inhale } \text{rename}_t(Q); \text{ dep} \text{ ++ } 1) \\ &= \text{sem}B_M(\{\bar{h}_{\vec{y}}(|\varphi|)\} \oplus h(y) \oplus \text{deps}(n) \oplus \text{rename}_t(\text{Inh}(Q)), \text{ dep} \text{ ++ } 1) \\ &= \{\bar{h}_{\vec{y}}(|\varphi|)\} \oplus h(y) \oplus \text{deps}(n+1) \oplus \text{rename}_t(\text{Inh}(Q)) \\ &= \{\bar{h}_{\vec{y}}(|\varphi|)\} \oplus h(y) \oplus \text{deps}(n+1) \oplus \text{rename}_t(\text{Inh}(Q_{\text{ind}})) \quad (\text{Using Equations 18, 20, 21, and } \vec{x} \cap \vec{y} = \emptyset) \\ &= \{\bar{h}_{\vec{y}}(|\varphi|)\} \oplus h(y) \oplus \text{deps}(n+1) \oplus \pi_{\vec{x} \cup \vec{y}}^{\vec{x} \cup \vec{y}}(\text{sem}_M(A, \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s')))) \quad (\text{Using Equation 23}) \\ &= \{\bar{h}_{\vec{y}}(|\varphi|)\} \oplus \pi_{\vec{x} \cup \vec{y}}^{\vec{x} \cup \vec{y}}(\text{sem}_M(A, \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s')))) \oplus \text{deps}(n+1) \end{aligned}$$

Therefore

$$\begin{aligned} & \pi_{\vec{V}}(\text{sem}_M(A \oplus \text{deps}(n+1), \text{dep} \text{ -- } 1; \vec{y} := m(\vec{x}); \text{ dep} \text{ ++ } 1)) \\ &= \bigcup_{\varphi \in A} (\pi_{\vec{V}}(\{\bar{h}_{\vec{y}}(|\varphi|)\} \oplus \pi_{\vec{x} \cup \vec{y}}^{\vec{x} \cup \vec{y}}(\text{sem}_M(A, \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s'))))) \\ &= \bigcup_{\varphi \in A} (\{\pi_{\vec{V}-\vec{y}}(|\varphi|)\} \oplus \pi_{\vec{x} \cup \vec{y}}^{\vec{x} \cup \vec{y}}(\text{sem}_M(A, \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s')))) \\ &= \pi_{\vec{V}-\vec{y}}(|A|) \oplus \pi_{\vec{x} \cup \vec{y}}^{\vec{x} \cup \vec{y}}(\text{sem}_M(A, \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s')))) \\ &= \pi_{\vec{V}}(\text{sem}_M(A, \text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s')))) \quad (\text{Since } \vec{V} \cap \text{modif}(\text{inl}_M^n(\vec{U} \cup \text{modif}(s'), s')) \subseteq \vec{y}) \\ &= \pi_{\vec{V}}(\text{sem}_M(A, \text{inl}_M^{n+1}(\vec{U}, \vec{y} := m(\vec{x})))) \end{aligned}$$

10.2 Proving point 3

Since $\text{collectAn}_M^{n+1}(\vec{U}, \vec{V}, A, \vec{y} := m(\vec{x})) = \{(m, \text{ind}, P_{\text{ind}}, Q_{\text{ind}})\} \cup \text{collectAn}_M^n(\vec{U} \cup \text{modif}(s'), \text{read}(s'), A, s')$, we have

$$\begin{aligned} & \text{verMethodsB}(\text{annot}(M, \text{collectAn}_M^{n+1}(\vec{U}, \vec{V}, A, \vec{y} := m(\vec{x})))) \iff \\ & \text{ver}B_M(h(\text{args}) \oplus h(\text{rets}), \text{inhale } P_{\text{ind}}; s; \text{ exhale } Q_{\text{ind}}) \wedge \text{verMethodsB}(\text{annot}(M, \text{collectAn}_M^n(\vec{U} \cup \text{modif}(s'), \text{read}(s'), A, s')))) \end{aligned}$$

The first part is given by Equation 14, thus we only need to prove the second part.

By inverting the renaming (which is well-formed), we have

$$\begin{aligned}
& verB_M(h(args) \oplus h(rets), \mathbf{inhale} P_{ind} ; s ; \mathbf{exhale} Q_{ind}) \\
& \iff verB_M(h(rename_t(\overrightarrow{args})) \oplus h(rename_t(\overrightarrow{rets})), \mathbf{inhale} rename_t(P_{ind}) ; rename_t(s) ; \mathbf{exhale} rename_t(Q_{ind})) \\
& \iff verB_M(h(\overrightarrow{x}) \oplus h(\overrightarrow{y}), \mathbf{inhale} rename_t(P_{ind}) ; s' ; \mathbf{exhale} rename_t(Q_{ind})) \\
& \iff verB_M(h(\overrightarrow{x}) \oplus h(\overrightarrow{y}) \oplus rename_t(Inh(P_{ind})), s' ; \mathbf{exhale} rename_t(Q_{ind})) \\
& \iff verB_M(h(\overrightarrow{x}) \oplus h(\overrightarrow{y}) \oplus \pi_{\overrightarrow{x} \cup \overrightarrow{y}}(A) \oplus deps(n), s' ; \mathbf{exhale} rename_t(Q_{ind})) \quad (\text{Using Equation 22}) \\
& \iff verB_M(\pi_{\overrightarrow{x} \cup \overrightarrow{y}}(A) \oplus deps(n), s' ; \mathbf{exhale} rename_t(Q_{ind})) \\
& \iff verB_M(\pi_{\overrightarrow{x} \cup \overrightarrow{y}}(A) \oplus deps(n), s') \wedge semB_M(\pi_{\overrightarrow{x} \cup \overrightarrow{y}}(A) \oplus deps(n), s') \succeq rename_t(Inh(Q_{ind}))) \quad (24)
\end{aligned}$$

Since $\sigma(A) \cap read(s') \subseteq \overrightarrow{x} \cup \overrightarrow{y}$ (renaming property), we have

$$verB_M(\pi_{\overrightarrow{x} \cup \overrightarrow{y}}(A) \oplus deps(n), s') \iff verB_M(A \oplus deps(n), s')$$

given by Equation 14, which concludes the left part of Equation 24.

Moreover

$$\begin{aligned}
& \pi_{read(s')} (semB_M(\pi_{\overrightarrow{x} \cup \overrightarrow{y}}(A) \oplus deps(n), s')) \\
& = \pi_{\overrightarrow{x} \cup \overrightarrow{y}} (semB_M(A \oplus deps(n), s')) \quad (\text{Since } \overrightarrow{x} \cup \overrightarrow{y} \subseteq read(s')) \\
& = \pi_{\overrightarrow{x} \cup \overrightarrow{y}} (\pi_{read(s')} (semB_M(A \oplus deps(n), s'))) \quad (\text{Since } \overrightarrow{x} \cup \overrightarrow{y} \subseteq read(s')) \\
& = \pi_{\overrightarrow{x} \cup \overrightarrow{y}} (\pi_{read(s')} (sem_M(A, inl_M^n(\overrightarrow{U} \cup \text{modif}(s'), s')))) \quad (\text{Using Equation 15}) \\
& = \pi_{read(s')} (\pi_{\overrightarrow{x} \cup \overrightarrow{y}} (sem_M(A, inl_M^n(\overrightarrow{U} \cup \text{modif}(s'), s')))) \\
& = \pi_{read(s')} (rename_t(Inh(Q_{ind}))) \quad (\text{Using Equation 23}) \\
& = rename_t(Inh(Q_{ind}))
\end{aligned}$$

Therefore

$$semB_M(\pi_{\overrightarrow{x} \cup \overrightarrow{y}}(A) \oplus deps(n), s') \succeq \pi_{read(s')} (semB_M(\pi_{\overrightarrow{x} \cup \overrightarrow{y}}(A) \oplus deps(n), s')) = rename_t(Inh(Q_{ind}))$$

which concludes Equation 24. \square

11 Induction case: Loop ($depth = n + 1$)

Lemma 10. *If*

1. $CompletenessInv_M^n(s)$
2. $CompletenessInv_M^n(\mathbf{while} (b) \mathbf{inv} I \{s\})$

then

$$CompletenessInv_M^{n+1}(\mathbf{while} (b) \mathbf{inv} I \{s\})$$

Proof. Let $w := \mathbf{while} (b) \mathbf{inv} I \{s\}$, we assume $CompletenessInv_M^n(s)$ and $CompletenessInv_M^n(w)$.

Let $\varphi \in A$, $\vec{l} := \text{modif}(s) \cap \sigma(\varphi)$, $s' := inl_M^n(\vec{U}, s)$, and $\vec{U}' := \vec{U} \cup read(s')$. To prove the invariant, we need to prove the following points:

1. $verB_M(\{\varphi\} \oplus deps(n+1), \mathbf{havoc} \vec{l} ; \mathbf{assign} 1 \leq dep \leq n+1 ; \mathbf{inhale} I ; \mathbf{assume} b ; dep \leftarrow 1 ; s ; \mathbf{exhale} I)$
2. $ver_M(\{\varphi\} \oplus deps(n+1), \mathbf{exhale} I ; \mathbf{havoc} \vec{l} ; \mathbf{assign} 0 \leq dep \leq n+1 ; \mathbf{inhale} I ; \mathbf{assume} \neg b ; dep \leftarrow n+1)$
3. $\pi_{\vec{V}}(sem_M(A, inl_M^{n+1}(\vec{U}', \mathbf{while} (b) \mathbf{inv} I \{s\}))) = \pi_{\vec{V}}(semB_M(A \oplus deps(n+1), \mathbf{while} (b) \mathbf{inv} I \{s\}))$
4. $verMethodsB(annot(M, collectAn_M^{n+1}(\vec{U}, \vec{V}, A, \mathbf{while} (b) \mathbf{inv} I \{s\})))$

Points 1 and 2 combined yield $verB_M(A \oplus deps(n), \mathbf{while} (b) \mathbf{inv} I \{s\})$.

By definition, we have

$$inl_M^{n+1}(\vec{U}, \mathbf{while} (b) \mathbf{inv} I \{s\}) = \mathbf{if} (*) \{ \mathbf{assume} b ; inl_M^n(\vec{U}, s) ; inl_M^n(\vec{U}', \mathbf{while} (b) \mathbf{inv} I \{s\}) \} \mathbf{else} \{ \mathbf{assume} \neg b \}$$

We thus have $ver_M(A, \mathbf{assume} b ; inl_M^n(\vec{U}, s) ; inl_M^n(\vec{U}', w))$, which gives us $ver_M(f_b(A), inl_M^n(\vec{U}, s))$.

From $CompletenessInv_M^n(s)$, we get

$$verB_M(f_b(A) \oplus deps(n), s) \quad (25)$$

$$\wedge \pi_{\vec{V}}(sem_M(f_b(A), inl_M^n(\vec{U}, s))) = \pi_{\vec{V}}(semB_M(f_b(A) \oplus deps(n), s)) \quad (26)$$

$$\wedge verMethodsB(annot(M, collectAn_M^n(\vec{U}, \vec{V}, A, s))) \quad (27)$$

Let $s' := \text{inl}_M^n(\vec{U}, s)$, and $A' := \text{sem}_M(f_b(A), s')$. Using *CompletenessInv* $_M^n(\vec{U}', w)$, we get

$$\text{ver}B_M(A' \oplus \text{deps}(n), w) \quad (28)$$

$$\wedge \pi_{\vec{V}}(\text{sem}_M(A', \text{inl}_M^n(\vec{U}', w))) = \pi_{\vec{V}}(\text{sem}B_M(A' \oplus \text{deps}(n), w)) \quad (29)$$

$$\wedge \text{verMethods}B(\text{annot}(M, \text{collectAn}_M^n(\vec{U}', \vec{V}, A', w))) \quad (30)$$

11.1 Proving point 4

By definition, we have

$$\text{collectAn}_M^{n+1}(\vec{U}, \vec{V}, A', \text{while } (b) \text{ inv } I \{s\}) = \text{collectAn}_M^n(\vec{U}, \vec{V}, A, s) \cup \text{collectAn}_M^n(\vec{U}', \vec{V}, A', \text{while } (b) \text{ inv } I \{s\})$$

Combined with Equation 27 and Equation 30, we get **point 3**.

11.2 Dividing point 1

We have

$$\begin{aligned} & \text{ver}B_M(\{|\varphi|\} \oplus \text{deps}(n+1), \text{havoc } \vec{l}; \text{assign } 1 \leq \text{dep} \leq n+1; \text{inhale } I; \text{assume } b; \text{dep} \leftarrow 1; s; \text{exhale } I) \\ \iff & \text{ver}B_M(\{|\varphi|\} \oplus \text{deps}(n+1), \text{havoc } \vec{l}; \text{assign } \text{dep} = n+1; \text{inhale } I; \text{assume } b; \text{dep} \leftarrow 1; s; \text{exhale } I) \end{aligned} \quad (\text{Point 1.1})$$

$$\wedge \text{ver}B_M(\{|\varphi|\} \oplus \text{deps}(n+1), \text{havoc } \vec{l}; \text{assign } 1 \leq \text{dep} \leq n; \text{inhale } I; \text{assume } b; \text{dep} \leftarrow 1; s; \text{exhale } I) \quad (\text{Point 1.2})$$

11.3 Proving point 1.1

We have

$$\begin{aligned} & \text{sem}B_M(\{|\varphi|\} \oplus \text{deps}(n+1), \text{havoc } \vec{l}; \text{assign } \text{dep} = n+1; \text{inhale } I) \\ = & \{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus \text{deps}(n+1) \oplus \text{Inh}(I) \quad (\text{Normal semantics}) \\ = & \{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus \text{deps}(n+1) \oplus \pi_{\vec{V}}(A) \quad (\text{Maximally annotated hypothesis}) \end{aligned}$$

It follows that

$$\pi_{\vec{V}}(\text{sem}B_M(\{|\varphi|\} \oplus \text{deps}(n+1), \text{havoc } \vec{l}; \text{assign } \text{dep} = n+1; \text{inhale } I)) \subseteq \pi_{\vec{V}}(A) \oplus \text{deps}(n+1)$$

Then

$$\pi_{\vec{V}}(\text{sem}B_M(\{|\varphi|\}, \text{havoc } \vec{l}; \text{assign } \text{dep} = n+1; \text{inhale } I; \text{assume } b; \text{dep} \leftarrow 1)) \subseteq \pi_{\vec{V}}(f_b(A)) \oplus \text{deps}(n)$$

From Equation 25 and since $\text{read}(s) \subseteq \vec{V}$, we get:

$$\text{ver}B_M(\{|\varphi|\} \oplus \text{deps}(n+1), \text{havoc } \vec{l}; \text{assign } \text{dep} = n+1; \text{inhale } I; \text{assume } b; \text{dep} \leftarrow 1; s)$$

and

$$\begin{aligned} & \text{sem}B_M(\{|\varphi|\} \oplus \text{deps}(n+1), \text{havoc } \vec{l}; \text{assign } \text{dep} = n+1; \text{inhale } I; \text{assume } b; \text{dep} \leftarrow 1; s) \\ \succeq & \pi_{\vec{V}}(\text{sem}B_M(\{|\varphi|\} \oplus \text{deps}(n+1), \text{havoc } \vec{l}; \text{assign } \text{dep} = n+1; \text{inhale } I; \text{assume } b; \text{dep} \leftarrow 1; s)) \\ = & \pi_{\vec{V}}(\text{sem}B_M(\text{sem}B_M(\{|\varphi|\} \oplus \text{deps}(n+1), \text{havoc } \vec{l}; \text{assign } \text{dep} = n+1; \text{inhale } I; \text{assume } b; \text{dep} \leftarrow 1), s)) \\ = & \text{sem}B_M(\pi_{\vec{V}}(\text{sem}B_M(\{|\varphi|\} \oplus \text{deps}(n+1))), \text{havoc } \vec{l}; \text{assign } \text{dep} = n+1; \text{inhale } I; \text{assume } b; \text{dep} \leftarrow 1), s) \\ & \quad (\text{Since } \text{read}(s) \subseteq \vec{V}) \\ \succeq & \text{sem}B_M(\pi_{\vec{V}}(f_b(A) \oplus \text{deps}(n)), s) \\ = & \pi_{\vec{V}}(\text{sem}B_M(f_b(A) \oplus \text{deps}(n), s)) \\ = & \pi_{\vec{V}}(\text{sem}_M(f_b(A, \text{inl}_M^n(\vec{U}, s)))) \quad (\text{Using Equation 26}) \\ = & \text{Inh}(I) \quad (\text{Maximally annotated hypothesis}) \end{aligned}$$

This concludes the case, by definition of the verification of **exhale** statements. Therefore, it remains to prove **point 1.2** to have **point 1**.

11.4 Dividing point 2

For all $\varphi \in A$:

$$\begin{aligned} & \text{ver}_M(\{\varphi\} \oplus \text{deps}(n+1), \text{exhale } I; \text{havoc } \vec{l}; \text{assign } 0 \leq \text{dep} \leq n+1; \text{inhale } I; \text{assume } \neg b; \text{dep} \leftarrow n+1) \\ \iff & \text{ver}_M(\{\varphi\} \oplus \text{deps}(n+1), \text{exhale } I; \text{havoc } \vec{l}; \text{assign } \text{dep} = n+1; \text{inhale } I; \text{assume } \neg b; \text{dep} \leftarrow n+1) \end{aligned} \quad (\text{Point 2.1})$$

$$\wedge \text{ver}_M(\{\varphi\} \oplus \text{deps}(n+1), \text{exhale } I; \text{havoc } \vec{l}; \text{assign } 0 \leq \text{dep} \leq n; \text{inhale } I; \text{assume } \neg b; \text{dep} \leftarrow n+1) \quad (\text{Point 2.2})$$

11.5 Dividing point 3

We want to show

$$\pi_{\vec{V}}(\text{sem}_M(A, \text{inl}_M^{n+1}(\vec{U}, \text{while } (b) \text{ inv } I \{s\}))) = \pi_{\vec{V}}(\text{sem}_M(A \oplus \text{deps}(n+1), \text{while } (b) \text{ inv } I \{s\}))$$

We have

$$\begin{aligned} \text{sem}_M(A, \text{inl}_M^{n+1}(\vec{U}, \text{while } (b) \text{ inv } I \{s\})) &= \text{sem}_M(A, \text{if } (*) \{ \text{assume } b ; \text{inl}_M^n(\vec{U}, s) ; \text{inl}_M^n(\vec{U}', w) \} \text{ else } \{ \text{assume } \neg b \}) \\ &= \text{sem}_M(f_b(A), \text{inl}_M^n(\vec{U}, s) ; \text{inl}_M^n(\vec{U}', w)) \cup f_{\neg b}(A) \end{aligned} \quad (31)$$

Using

$$\forall \varphi \in A. \text{sem}_M(\{\varphi\} \oplus \text{deps}(n+1), \text{exhale } I ; \text{havoc } \vec{l}) = \{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus \text{deps}(n+1)$$

we obtain

$$\begin{aligned} &\text{sem}_M(A \oplus \text{deps}(n+1), w) \\ &= \text{sem}_M(A \oplus \text{deps}(n+1), \text{exhale } I ; \text{havoc } \vec{l} ; \text{assign } 0 \leq \text{dep} \leq n+1 ; \text{inhale } I ; \text{assume } \neg b ; \text{dep} \leftarrow n+1) \\ &= \bigcup_{\varphi \in A} \text{sem}_M(\{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus \text{deps}(n+1), \text{assign } \text{dep} = n+1 ; \text{inhale } I ; \text{assume } \neg b ; \text{dep} \leftarrow n+1) \quad (S_1) \\ &\cup \bigcup_{\varphi \in A} \text{sem}_M(\{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus \text{deps}(n+1), \text{assign } 0 \leq \text{dep} \leq n ; \text{inhale } I ; \text{assume } \neg b ; \text{dep} \leftarrow n+1) \quad (S_2) \end{aligned}$$

We will prove the following points:

- **Point 3.1:** $\pi_{\vec{V}}(S_1) = \pi_{\vec{V}}(f_{\neg b}(A))$
- **Point 3.2:** $\pi_{\vec{V}}(S_2) = \pi_{\vec{V}}(\text{sem}_M(A', \text{inl}_M^n(\vec{U}', w)))$

Using these two points, we get

$$\begin{aligned} \pi_{\vec{V}}(\text{sem}_M(A, w)) &= \pi_{\vec{V}}(S_1) \cup \pi_{\vec{V}}(S_2) \\ &= \pi_{\vec{V}}(f_{\neg b}(A)) \cup \pi_{\vec{V}}(\text{sem}_M(A', \text{inl}_M^n(\vec{U}', w))) \\ &= \pi_{\vec{V}}(f_{\neg b}(A)) \cup \pi_{\vec{V}}(\text{sem}_M(f_b(A), \text{inl}_M^n(\vec{U}, s) ; \text{inl}_M^n(\vec{U}', w))) \\ &= \pi_{\vec{V}}(\text{sem}_M(A, \text{inl}_M^{n+1}(\vec{U}', w))) \end{aligned}$$

11.6 Proving points 2.1 and 3.1

Let $\varphi \in A$.

$$\begin{aligned} &\text{sem}_M(\{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus \text{deps}(n+1), \text{assign } \text{dep} = n+1 ; \text{inhale } I) \\ &= \{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus \text{deps}(n+1) \oplus \text{Inh}(I) \\ &= \{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus \text{deps}(n+1) \oplus \pi_{\vec{V}}(A) \end{aligned} \quad (\text{Maximally annotated hypothesis})$$

Then

$$\pi_{\vec{V}}(\text{sem}_M(\{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus \text{deps}(n+1), \text{assign } \text{dep} = n+1 ; \text{inhale } I)) = \pi_{\vec{V}}(A)$$

We then have

$$\pi_{\vec{V}}(\{\varphi\}) \subseteq \pi_{\vec{V}}(\text{sem}_M(\{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus \text{deps}(n+1), \text{assign } \text{dep} = n+1 ; \text{inhale } I)) \subseteq \pi_{\vec{V}}(A)$$

Thus, we get

$$\pi_{\vec{V}}(\text{sem}_M(A \oplus \text{deps}(n+1), \text{exhale } I ; \text{havoc } \vec{l} ; \text{assign } 0 \leq \text{dep} \leq n+1 ; \text{inhale } I)) = \pi_{\vec{V}}(A)$$

Then, since $\text{dep} \notin \vec{V}$ and $\text{read}(\text{assume } \neg b) \subseteq \vec{V}$,

$$\pi_{\vec{V}}(S_1) = \pi_{\vec{V}}(f_{\neg b}(A))$$

which proves **Point 3.1**

To prove points 1.2, 2.2, 3.2, we need to distinguish two cases.

11.7 First case: $A' \neq \emptyset$

In this case, there exists $\varphi' \in A'$. Since loop bodies cannot create variables (hypothesis), $\sigma(\varphi) = \sigma(\varphi')$. Moreover, by definition of A' and of \vec{l} , we have that φ and φ' are equal outside of \vec{l} , namely $\bar{h}_{\vec{l}}(|\varphi|) = \bar{h}_{\vec{l}}(|\varphi'|)$

11.7.1 Proving point 1.2

We have $semB_M(\{|\varphi|\} \oplus deps(n+1), \mathbf{havoc} \vec{l}) = semB_M(\{|\varphi'|\} \oplus deps(n+1), \mathbf{havoc} \vec{l})$. We can conclude using $verB_M(A' \oplus deps(n+1), w)$ (Equation 28).

11.7.2 Proving point 2.2 and 3.2

Since an iteration cannot create new variables, we have

$$\{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) = \{\bar{h}_{\vec{l}}(|\varphi'|\}) \oplus h(\vec{l})$$

From Equation 28, we get

$$\forall \varphi \in A.ver_M(\{\varphi\}, \mathbf{exhale} I ; \mathbf{havoc} \vec{l} ; \mathbf{assign} 0 \leq dep \leq n ; \mathbf{inhale} I ; \mathbf{assume} \neg b ; dep \leftarrow n+1)$$

and from Equation 29, we get

$$\pi_{\vec{V}}(S_2) = \pi_{\vec{V}}(sem_M(A', inl_M^n(\vec{U}', w)))$$

11.8 Second case: $A' = \emptyset$

By induction over $maxAnnot$, we get:

$$\forall i \leq n. \bar{h}_{\vec{V}}(|\varphi|) \oplus h(\vec{V}) \oplus deps(i) \oplus Inh(I) = \emptyset$$

Since $\vec{l} \subseteq \text{modif}(s) \subseteq \text{read}(s) \subseteq \vec{V}$, we have that $\bar{h}_{\vec{l}}(|\varphi|) \oplus h(\vec{l}) \subseteq \bar{h}_{\vec{V}}(|\varphi|) \oplus h(\vec{V})$, thus

$$\forall i \leq n. \bar{h}_{\vec{l}}(|\varphi|) \oplus h(\vec{l}) \oplus deps(i) \oplus Inh(I) = \emptyset$$

Points 1.2 and 2.2 follow immediately.

11.8.1 Proving point 3.2

:

$$\begin{aligned} & \pi_{\vec{V}}(S_2) \\ &= \pi_{\vec{V}}\left(\bigcup_{\varphi \in A} sem_M(\{\bar{h}_{\vec{l}}(|\varphi|)\} \oplus h(\vec{l}) \oplus deps(n+1), \mathbf{assign} 0 \leq dep \leq n ; \mathbf{inhale} I ; \mathbf{assume} \neg b ; dep \leftarrow n+1)\right) \\ &= \pi_{\vec{V}}\left(\bigcup_{\varphi \in A} \emptyset\right) \\ &= \pi_{\vec{V}}(\emptyset) \\ &= \pi_{\vec{V}}(sem_M(\emptyset, inl_M^n(\vec{U}', w))) \\ &= \pi_{\vec{V}}(sem_M(A', inl_M^n(\vec{U}', w))) \end{aligned}$$

□