On the Design of Program Logics

Patrick Cousot[1][0000−0003−0101−9953]

Courant Institute of Mathematical Sciences, New York University
`pcousot@cims.nyu.edu` https://cs.nyu.edu/~pcousot/

Abstract. We illustrate the sound and complete construction of program logics by abstraction of a relational semantics into the theory of the logic followed by the formal derivation of the proof system. We consider Hoare logic, O'Hearn incorrectness logic and Hoare incorrectness logic formalizing debugging.

Keywords: Hoare logic · reverse Hoare logic · incorrectness logic · Hoare incorrectness logic · debugging · abstract interpretation.

1 Introduction

We have recently proposed a methodology [6] to construct sound and complete Hoare'style (or transformational) program logics by defining the structural relational semantics of the programming language, which is abstracted into the theory of the logic (specifying the true formulas of the logic) and then expressed as an equivalent proof system using Peter Aczel correspondence between set-theoretic fixpoints and deductive rules as well as fixpoint induction principles to handle iteration. The construction of the logic consists in

1. Defining the relational semantics $[\![\mathtt{S}]\!]$ of the programming language (in structural fixpoint form);
2. Defining the theory of the logics as an abstraction $\alpha(\{[\![\mathtt{S}]\!]\})$ of the collecting semantics $\{[\![\mathtt{S}]\!]\}$ (which is the strongest (hyper) property of statement $\mathtt{S}$);
3. Calculating the theory $\alpha(\{[\![\mathtt{S}]\!]\})$ in structural fixpoint form by fixpoint abstraction;
4. Calculating the proof system by fixpoint induction and Aczel correspondence [1] between fixpoints and deductive systems.

2 The Structural Natural Relational Semantics

We consider an imperative language $\mathbb{S}$ with assignments, sequential composition, conditionals, and conditional iteration. The syntax is $\mathtt{S} \in \mathbb{S} ::= \mathtt{x = A} \mid \mathtt{skip} \mid \mathtt{S;S} \mid \mathtt{if\ (B)\ S\ else\ S} \mid \mathtt{while\ (B)\ S}$. States $\sigma \in \Sigma \triangleq \mathbb{X} \to \mathbb{V}$ (also called environments) map variables $\mathtt{x} \in \mathbb{X}$ to their values $\sigma(\mathtt{x})$ in $\mathbb{V}$. We deliberately leave unspecified the syntax and semantics of arithmetic expressions $\mathcal{A}[\![\mathtt{A}]\!] \in \Sigma \to \mathbb{V}$ and Boolean expressions $\mathcal{B}[\![\mathtt{B}]\!] \in \wp(\Sigma) \simeq \Sigma \to \{\mathsf{true}, \mathsf{false}\}$. The only assumption on expressions is the absence of side effects. The natural (or angelic) semantics $⟦\texttt{S}⟧ \in \wp(\Sigma \times \Sigma)$ ignores nontermination [14]. We use judgements $\sigma \vdash \texttt{S} \Rightarrow \sigma'$ for $\langle\sigma,\, \sigma'\rangle \in ⟦\texttt{S}⟧$. In addition we write $\sigma \vdash \texttt{while (B) S} \overset{i}{\Rightarrow} \sigma'$ to mean that if σ is a state before executing `while (B) S`, then σ' is reachable after 0 or more iterations of the loop body (so $\sigma = \sigma'$ for 0 iterations, before entering the loop in case (1.a)). The semantics of iteration $\texttt{W} = \texttt{while (B) S}$ is

$$\text{(a) } \sigma \vdash \texttt{W} \overset{i}{\Rightarrow} \sigma \quad \text{(b) } \frac{\sigma \vdash \texttt{W} \overset{i}{\Rightarrow} \sigma',\ \mathcal{B}⟦\texttt{B}⟧\sigma',\ \sigma' \vdash \texttt{S} \Rightarrow \sigma''}{\sigma \vdash \texttt{W} \overset{i}{\Rightarrow} \sigma''} \quad \text{(c) } \frac{\sigma \vdash \texttt{W} \overset{i}{\Rightarrow} \sigma',\ \mathcal{B}⟦\neg\texttt{B}⟧\sigma'}{\sigma \vdash \texttt{W} \Rightarrow \sigma'} \quad (1)$$

Aczel correspondence between deductive systems and set-theoretic fixpoints [1] allows us to derive an equivalent fixpoint definition of the semantics.

$$F(X) \triangleq \mathsf{id} \cup (X \,⨾\, ⟦\texttt{B}⟧ \,⨾\, ⟦\texttt{S}⟧), \quad X \in \wp(\Sigma \times \Sigma) \quad (2)$$

$$⟦\texttt{while (B) S}⟧ \triangleq \mathsf{lfp}^{\subseteq} F \,⨾\, ⟦\neg\texttt{B}⟧ \quad (3)$$

where id is the identity relation and ⨾ is the composition of relations. The transformer F are defined on the complete lattice $\langle\wp(\Sigma \times \Sigma),\, \subseteq,\, \emptyset,\, \Sigma \times \Sigma,\, \cup,\, \cap\rangle$ and is $\subseteq$-increasing, so $\mathsf{lfp}^{\subseteq} F$ does exist [19].

3 Aczel correspondence between deductive systems and fixpoints

Rules $\frac{P}{c}$ on a universe $\mathcal{U}$ have a premise $P \in \wp(\mathcal{U})$ and a conclusion $c \in \mathcal{U}$. The premise $P = \emptyset$ is empty for axioms. A deductive system is a set of rules $R = \left\{\frac{P_i}{c_i} \;\middle|\; i \in \Delta\right\} \in \wp(\wp(\mathcal{U}) \times \mathcal{U})$. The semantics of $⦃R⦄$ is the subset of the universe $\mathcal{U}$ defined by the rules.

The traditional proof-theoretic semantics of deductive systems is the set of provable terms (called theorems). Therefore $⦃R⦄^p = \{t_n \in \mathcal{U} \mid \exists t_1, \ldots, t_{n-1} \in \mathcal{U} \,.\, \forall k \in [1, n] \,.\, \exists\frac{P}{c} \in R \,.\, P \subseteq \{t_1, \ldots, t_{k-1}\} \wedge t_k = c\}$ (such finite proofs require the premiss P to be finite but transfinite proofs are also possible [15, Chapter 11], [1, Definition 1.4.1]). The model-theoretic semantics of deductive systems is the least fixpoint $⦃R⦄^m = \mathsf{lfp}^{\subseteq} F_R$ where the consequence operator $F_R(X) \triangleq \{c \mid \exists\frac{P}{c} \in R \,.\, P \subseteq X\}$ is increasing. Peter Aczel [1] proves that $⦃R⦄^m = ⦃R⦄^p$. In the case of finite premisses, the idea is that the n-th iterates of F_R from $\emptyset$ is the set of all proofs of length n (hence the axioms for the first iterates). Conversely, any set-theoretic fixpoint $\mathsf{lfp}^{\subseteq} F$ of an increasing operator F (continuous for finite premisses) on the powerset $\wp(\mathcal{U})$ of a set $\mathcal{U}$ is the semantics $⦃R_F⦄^m = ⦃R_F⦄^p$ of a deductive system $R_F = \left\{\frac{P}{c} \;\middle|\; P \in \wp(\mathcal{U}) \wedge c \in F(P)\right\}$ (or $\left\{\frac{P}{c} \;\middle|\; P \in \wp(\mathcal{U}) \wedge c \in F(P) \wedge \forall P' \in \wp(\mathcal{U}) \,.\, c \in F(P') \Rightarrow P \subseteq P'\right\}$ to eliminate redundant rules). Given the fixpoint characterization of the theory of a logic, obtained by abstraction $\alpha(\{⟦\texttt{S}⟧\})$ of the fixpoint semantics $⟦\texttt{S}⟧$, we can derive the proof system of the logic using this correspondence between fixpoints and rules.

4 Abstractions

Since the strongest property $\{⟦\texttt{S}⟧\}$ of (the semantics $⟦\texttt{S}⟧$ of) a statement `S` is an hyperproperty, it must be abstracted into an execution property $\alpha_{\mathrm{C}}(\{⟦\texttt{S}⟧\}) = ⟦\texttt{S}⟧$

where $\alpha_{\mathrm{C}}(P) \triangleq \bigcup P$ is surjective and $\gamma_{\mathrm{C}}(S) \triangleq \wp(S)$ is injective and is a Galois retraction $\langle \wp(\wp(\mathcal{D})), \subseteq\rangle \xleftarrow[\alpha_{\mathrm{C}}]{\gamma_{\mathrm{C}}} \langle \wp(\mathcal{D}), \subseteq\rangle$ [5, Chapter 11]. So $[\![\mathtt{S}]\!]$ is the strongest property of executions of $\mathtt{S}$.

Then program logics check reachability from preconditions or accessibility of postconditions which can be formalized by the post-image isomorphism $\mathsf{post}([\![\mathtt{S}]\!])$ of the relational semantics where $\mathsf{post}(r)X \triangleq \{y \mid \exists x \in X \,.\, \langle x,\ y\rangle \in r\}$ maps a relation $r \in \wp(\mathcal{X} \times \mathcal{Y})$ to its union-preserving right-image $\mathsf{post}(r)$ so that $\langle \wp(\mathcal{X}), \subseteq\rangle \xleftarrow[\mathsf{post}(r)]{\widetilde{\mathsf{pre}}(r)} \langle \wp(\mathcal{Y}),\ \subseteq\rangle$ is a Galois connection with $\widetilde{\mathsf{pre}}(r)Q = \neg\mathsf{post}(r^{-1})\neg Q = \{x \mid \forall y \,.\, \langle x,\ y\rangle \in r \Rightarrow y \in Q\}$.

Contrary to predicate transformers, program logics are not functions but relations between predicates. Since a function $f \in \mathcal{X} \to \mathcal{Y}$ is isomorphic to its graph $\alpha_{\mathrm{G}}(f) = \{\langle x,\ f(x)\rangle \mid x \in \mathcal{X}\}$, we can abstract a predicate transformer into a functional relation, which is a Galois isomorphism $\langle \mathcal{X} \to \mathcal{Y},\ =\rangle \xleftarrow[\alpha_{\mathrm{G}}]{\gamma_{\mathrm{G}}} \langle \wp_{\mathsf{fun}}(\mathcal{X} \times \mathcal{Y}),\ =\rangle$ where $\gamma_{\mathrm{G}}(r) \triangleq \boldsymbol{\lambda} x \bullet (y \textit{ such that } \langle x,\ y\rangle \in r)$ is uniquely well-defined since r is a functional relation.

5 The Strongest Postcondition Logic Theory

We can now define the strongest postcondition logic theory of a statement $\mathtt{S}$ as

$$\mathcal{T}[\![\mathtt{S}]\!] = \alpha_{\mathrm{G}} \circ \mathsf{post} \circ \alpha_{\mathrm{C}}(\{[\![\mathtt{S}]\!]\}) = \{\langle P,\ \mathsf{post}[\![\mathtt{S}]\!]P\rangle \mid P \in \wp(\Sigma)\} \tag{4}$$

The next step is to express this theory in fixpoint form. The main result from [9] is that the abstraction of a fixpoint is a fixpoint.

Theorem 1 (Fixpoint abstraction [9]). *If* $\langle C,\ \sqsubseteq\rangle \xleftarrow[\alpha]{\gamma} \langle A,\ \preceq\rangle$ *is a Galois connection between complete lattices* $\langle C,\ \sqsubseteq\rangle$ *and* $\langle A,\ \preceq\rangle$*,* $f \in C \xrightarrow{i} C$ *and* $\bar{f} \in A \xrightarrow{i} A$ *are increasing and commuting, that is,* $\alpha \circ f = \bar{f} \circ \alpha$*, then* $\alpha(\mathsf{lfp}^{\sqsubseteq} f) = \mathsf{lfp}^{\preceq} \bar{f}$ *(while semi-commutation* $\alpha \circ f \mathrel{\dot{\preceq}} \bar{f} \circ \alpha$ *implies* $\alpha(\mathsf{lfp}^{\sqsubseteq} f) \preceq \mathsf{lfp}^{\preceq} \bar{f}$*).*

As a simple application, we have the following corollary (which will allows us to define the predicate transformer for an iteration as the least fixpoint of a predicate transformer, as opposed to a (predicate transformer) transformer).

Corollary 1 (Pointwise abstraction). *Let* $\langle L,\ \sqsubseteq,\ \top,\ \sqcup\rangle$ *and* $\langle L',\ \sqsubseteq',\ \top',\ \sqcup'\rangle$ *be complete lattices. Assume that* $F \in (L \to L') \xrightarrow{i} (L \to L')$ *is increasing and that for all* $Q \in L$*,* $\bar{F}_Q \in L' \xrightarrow{i} L'$ *is increasing. Assume* $\forall Q \in L \,.\, \forall f \in L \to L' \,.\, F(f)Q = \bar{F}_Q(f(Q))$*. Then* $\forall Q \in L \,.\, (\mathsf{lfp}^{\dot{\sqsubseteq}'} F)Q = \mathsf{lfp}^{\sqsubseteq'} \bar{F}_Q$*.*

By calculational design we get a fixpoint definition of the theory of strongest postconditions logics (common to Hoare logic and incorrectness logic with no consequence rules at all). For the iteration $\mathtt{W} = \mathtt{while\ (B)\ S}$, we get $\mathcal{T}[\![\mathtt{W}]\!] \triangleq \{\langle P,\ \mathsf{post}[\![\neg\mathtt{B}]\!](\mathsf{lfp}^{\subseteq} F'_P)\rangle \mid P \in \wp(\Sigma)\}$ where $F'_P(X) = P \cup \mathsf{post}([\![\mathtt{B}]\!] \fatsemi [\![\mathtt{S}]\!])X$.

For the proof, we have the commutation

$$\begin{aligned}
& \mathsf{post}(F(X))P \\
= \;& \mathsf{post}(\mathsf{id} \cup (X \mathbin{⨾} [\![\mathtt{B}]\!] \mathbin{⨾} [\![\mathtt{S}]\!]))P && \text{≀def. (2) of } F\text{∫} \\
= \;& \mathsf{post}(\mathsf{id})P \cup \mathsf{post}(X \mathbin{⨾} [\![\mathtt{B}]\!] \mathbin{⨾} [\![\mathtt{S}]\!])P && \text{≀}\mathsf{post}\text{ preserves arbitrary set unions∫} \\
= \;& P \cup \mathsf{post}([\![\mathtt{B}]\!] \mathbin{⨾} [\![\mathtt{S}]\!])(\mathsf{post}(X)P) && \text{≀def. }\mathsf{post}\text{, } \mathsf{post}(X \mathbin{⨾} Y) = \mathsf{post}(Y) \circ \mathsf{post}(X)\text{∫} \\
= \;& F'_P(\mathsf{post}(X)P) && \text{≀def. } F'_P \triangleq \boldsymbol{\lambda} X \bullet P \cup \mathsf{post}([\![\mathtt{B}]\!] \mathbin{⨾} [\![\mathtt{S}]\!])X\text{, Q.E.D.∫}
\end{aligned}$$

It follows that

$$\begin{aligned}
& \mathsf{post}[\![\mathtt{W}]\!] \\
= \;& \boldsymbol{\lambda} P \bullet \mathsf{post}(\mathsf{lfp}^{\subseteq} F \mathbin{⨾} [\![\neg\mathtt{B}]\!])P && \text{≀by (3) and Church } \lambda\text{-notation∫} \\
= \;& \boldsymbol{\lambda} P \bullet \mathsf{post}[\![\neg\mathtt{B}]\!](\mathsf{post}(\mathsf{lfp}^{\subseteq} F)P) && \text{≀}\mathsf{post}(X \mathbin{⨾} Y) = \mathsf{post}(Y) \circ \mathsf{post}(X)\text{∫} \\
= \;& \mathsf{post}[\![\neg\mathtt{B}]\!](\mathsf{lfp}^{\subseteq} F'_P) && \text{≀commutation and corollary 1, Q.E.D.∫} \qquad (5)
\end{aligned}$$

The characterization $\mathcal{T}[\![\mathtt{S}]\!] = \alpha_{\mathrm{G}} \circ \mathsf{post} \circ \alpha_{\mathrm{C}}(\{[\![\mathtt{S}]\!]\}) = \alpha_{\mathrm{G}} \circ \mathsf{post}([\![\mathtt{S}]\!]) = \{\langle P,\ \mathsf{post}[\![\mathtt{S}]\!]P\rangle \mid P \in \wp(\Sigma)\}$ follows directly from the definitions of α_{C} and α_{G}.

6 Deduction as Approximation

Lacking a consequence rule, the strongest postcondition logic theory is very strong and would, e.g., require the use of the strongest invariant for iteration, which by experience [20,16,12,13] is inadequately constraining.

The consequence abstraction can be defined by the component wise approximation $\langle x',\ y'\rangle \sqsubseteq,\!\preceq \langle x,\ y\rangle \triangleq x' \sqsubseteq x \wedge y' \preceq y$. Then $\mathsf{post}(\supseteq,\subseteq) = \boldsymbol{\lambda} R \bullet \{\langle P,\ Q\rangle \mid \exists \langle P',\ Q'\rangle \in R \;.\; P \subseteq P' \wedge Q' \subseteq Q\}$ adds an over approximating consequence rule to the strongest postcondition logic theory to get the theory of Hoare logic

$$\mathcal{T}_{\mathrm{HL}}[\![\mathtt{S}]\!] \triangleq \mathsf{post}(\supseteq,\subseteq)(\mathcal{T}[\![\mathtt{S}]\!]) = \{\langle P,\ Q\rangle \mid \mathsf{post}[\![\mathtt{S}]\!]P \subseteq Q\} \tag{6}$$

while the $\subseteq$-dual $\mathsf{post}(\subseteq,\supseteq) = \boldsymbol{\lambda} R \bullet \{\langle P,\ Q\rangle \mid \exists \langle P',\ Q'\rangle \in R \;.\; P' \subseteq P \wedge Q \subseteq Q'\}$ adds an under approximating consequence rule to the strongest postcondition logic theory to get the theory of incorrectness logic.

$$\mathcal{T}_{\mathrm{IL}}[\![\mathtt{S}]\!] \triangleq \mathsf{post}(\subseteq,\supseteq)(\mathcal{T}[\![\mathtt{S}]\!]) = \{\langle P,\ Q\rangle \mid Q \subseteq \mathsf{post}[\![\mathtt{S}]\!]P\} \tag{7}$$

Proof (of (6) and (7)).

$$\begin{aligned}
- \;& \mathsf{post}(\supseteq,\subseteq)(\mathcal{T}[\![\mathtt{S}]\!]) \\
= \;& \{\langle P,\ Q\rangle \mid \exists \langle P',\ Q'\rangle \in \mathcal{T}[\![\mathtt{S}]\!] \;.\; \langle P',\ Q'\rangle \supseteq.\!\subseteq \langle P,\ Q\rangle\} && \text{≀def. }\mathsf{post}\text{∫} \\
= \;& \{\langle P,\ Q\rangle \mid \exists \langle P',\ Q'\rangle \in \mathcal{T}[\![\mathtt{S}]\!] \;.\; P \subseteq P' \wedge Q' \subseteq Q\} \\
& && \text{≀and component wise order } \supseteq.\!\subseteq\text{∫} \\
= \;& \{\langle P,\ Q\rangle \mid \exists \langle P',\ Q'\rangle \in \{\langle P'',\ \mathsf{post}[\![\mathtt{S}]\!]P''\rangle \mid P'' \in \wp(\Sigma)\} \;.\; P \subseteq P' \wedge Q' \subseteq Q\} \\
& && \text{≀def. (4) of } \mathcal{T}[\![\mathtt{S}]\!]\text{∫} \\
= \;& \{\langle P,\ Q\rangle \mid \exists P', Q', P'' \;.\; P' = P'' \wedge Q' = \mathsf{post}[\![\mathtt{S}]\!]P'' \wedge P \subseteq P' \wedge Q' \subseteq Q\}
\end{aligned}$$

$$\wr \text{def. } \in \wr$$

$$= \{\langle P, Q\rangle \mid \exists P' \,.\, P \subseteq P' \wedge \mathsf{post}[\![\mathtt{S}]\!]P' \subseteq Q\} \qquad \wr\text{def. } = \wr$$

$$= \{\langle P, Q\rangle \mid \mathsf{post}[\![\mathtt{S}]\!](P) \subseteq Q\}$$

$$\wr(\Rightarrow)\ \mathsf{post}[\![\mathtt{S}]\!] \text{ is increasing, } (\Leftarrow) \text{ taking } P' = P\text{, proving (6)}\wr$$

– (7) follows from (6) by $\subseteq$-order duality. □

At this point, we could use Aczel correspondence between fixpoints and rules to get sound and complete proof rules with common rules for statements and different consequence rules for Hoare and incorrectness logics.

This is not satisfactory since, for iteration, we can approximate the pre and postconditions by the consequence rules but not the loop invariant itself. Requiring the invariant to be the strongest would be too demanding. This is precisely the point of fixpoint induction, which enables the use of consequences for the loop invariant. Fixpoint induction differs for Hoare logic (which requires a fixpoint over approximation) and incorrectness logic (which requires a fixpoint under approximation).

7 Fixpoint induction

A sound and complete least fixpoint over approximation method is provided by David Park [18,4].

Theorem 2 (Least fixpoint over approximation [18]). *Let $\langle L, \sqsubseteq, \bot, \top, \sqcup, \sqcap\rangle$ be a complete lattice, $f \in L \xrightarrow{i} L$ be increasing, and $p \in L$. Then $\mathsf{lfp}^{\sqsubseteq} f \sqsubseteq p$ if and only if $\exists i \in L \,.\, f(i) \sqsubseteq i \wedge i \sqsubseteq p$.*

where the inductive invariant i is an over approximation of the strongest invariant $\mathsf{lfp}^{\sqsubseteq} f$.

For under approximation of least fixpoints, we can use the generalization [4] of Scott-Kleene induction based on transfinite induction when continuity does not apply and follows directly from the constructive version of Tarski's fixpoint theorem [8].

Theorem 3 (Fixpoint Under Approximation by Transfinite Iterates). *Let $f \in L \xrightarrow{i} L$ be an increasing function on a* CPO *$\langle L, \sqsubseteq, \bot, \sqcup\rangle$ (i.e. every increasing chain in L has a least upper bound in L, including $\bot = \sqcup\emptyset$). $P \in L$ is a fixpoint under approximation, i.e. $P \sqsubseteq \mathsf{lfp}^{\sqsubseteq} f$, if and only if there exists an increasing transfinite sequence $\langle X^\delta, \delta \in \mathbb{O}\rangle$ such that $X^0 = \bot$, $X^{\delta+1} \sqsubseteq f(X^\delta)$ for successor ordinals, $\bigsqcup_{\delta<\lambda} X^\delta$ exists for limit ordinals λ such that $X^\lambda \sqsubseteq \bigsqcup_{\delta<\lambda} X^\delta$, and $\exists \delta \in \mathbb{O} \,.\, P \sqsubseteq X^\delta$.*

Theorem 3 could have assumed the existence of a well-founded set $\langle W, \preccurlyeq\rangle \in \mathfrak{Wf}$ to replace the ordinals $\langle \mathbb{O}, \leqslant\rangle$. If f is continuous then $\delta = \omega$ is the first infinite ordinal.

Finally, to propose an alternative axiomatization of incorrectness, we will use the following theorem.

Theorem 4 (Non empty intersection with abstraction of least fixpoint). *Assume that (1) $\langle L, \sqsubseteq, \bot, \top, \sqcap, \sqcup\rangle$ is an atomic complete lattice; (2) $f \in L \rightarrow L$ preserves nonempty joins $\sqcup$; (3) $\langle L, \sqsubseteq\rangle \xleftarrow[\alpha]{\gamma} \!\!\!\twoheadrightarrow \langle \bar{L}, \preceq, \curlywedge\rangle$; (4) $\bar{Q} \in \bar{L} \setminus \{0\}$ where $0 \triangleq \alpha(\bot)$; (5) There exists an inductive invariant $I \in L$ of f (i.e. $f(I) \sqsubseteq I$); (6) $\langle W, \leqslant\rangle$ is a well-founded set and $\nu \in \mathsf{atoms}(I) \rightarrow W$ is a (variant) function; (7) There exists a sequence $\langle a_i \in \mathsf{atoms}(I),\ i \in [1, \infty]\rangle$ that (7.a) $a_1 \in f(\bot)$, (7.b) $\forall i \in [1, \infty]\ .\ a_{i+1} \in \mathsf{atoms}(f(a_i))$, (7.c) $\forall i \in [1, \infty]\ .\ (a_i \neq a_{i+1}) \Rightarrow (\nu(a_i) > \nu(a_{i+1})$, (7.d) $\forall i \in [1, \infty]\ .\ (\nu(a_i) \not> \nu(a_{i+1}) \Rightarrow \alpha(a_i) \curlywedge \bar{Q} \neq 0$; Then, hypotheses (1) to (7) imply $\alpha(\mathsf{lfp}^{\sqsubseteq} f) \curlywedge \bar{Q} \neq 0$. Conversely (1) to (4) and $\mathsf{lfp}^{\sqsubseteq} f \sqcap \gamma(\bar{Q}) \neq \bot$ imply (5) to (7).*

Notice that if $L = \wp(\Sigma)$ then $\mathsf{atoms}(L) = \{\{x\} \mid x \in L\}$ so that $I \in \wp(\Sigma)$ and ν can be chosen in $I \rightarrow W$ instead of $\{\{x\} \mid x \in I\} \rightarrow W$. The proof of theorems 3 and 4 is given in [7].

8 Calculational Design of Hoare Logic

The calculus is by structural induction i.e. on the program syntax. The only non-trivial case is iteration `W = while (B) S`. The theory of Hoare logic for iteration is

$$\begin{aligned}\mathcal{T}_{\mathrm{HL}}(\mathtt{W}) &\triangleq \mathsf{post}(\supseteq, \subseteq)(\mathcal{T}[\![\mathtt{W}]\!]) \qquad (8)\\ &= \{\langle P,\ Q\rangle \mid \exists I\ .\ P \subseteq I \wedge \langle I \cap \mathcal{B}[\![\mathtt{B}]\!],\ I\rangle \in T_{\mathrm{HL}}(\mathtt{S}) \wedge (I \cap \neg\mathcal{B}[\![\mathtt{B}]\!]) \subseteq Q\}\end{aligned}$$

Proof (of (8)).

$$\begin{aligned}
&\mathcal{T}_{\mathrm{HL}}[\![\mathtt{W}]\!]\\
=\ & \mathsf{post}(\supseteq.\subseteq)(\mathcal{T}[\![\mathtt{W}]\!]) && \text{≀def. (6) of } \mathcal{T}_{\mathrm{HL}}\text{≀}\\
=\ & \{\langle P',\ Q'\rangle \mid \exists\langle P,\ Q\rangle \in \mathcal{T}[\![\mathtt{W}]\!]\ .\ \langle P,\ Q\rangle \supseteq, \subseteq \langle P',\ Q'\rangle\} && \text{≀def. } \mathsf{post}\text{≀}\\
=\ & \{\langle P',\ Q'\rangle \mid \exists\langle P,\ Q\rangle \in \mathcal{T}[\![\mathtt{W}]\!]\ .\ P' \subseteq P \wedge Q \subseteq Q'\} && \text{≀component wise def. } \supseteq, \subseteq\text{≀}\\
=\ & \{\langle P',\ Q'\rangle \mid \exists P, Q\ .\ P' \subseteq P \wedge \mathsf{post}[\![\neg\mathtt{B}]\!](\mathsf{lfp}^{\subseteq} F'_P) \subseteq Q \wedge Q \subseteq Q'\} && \text{≀(5)≀}\\
=\ & \{\langle P',\ Q'\rangle \mid \exists P\ .\ P' \subseteq P \wedge \mathsf{post}[\![\neg\mathtt{B}]\!](\mathsf{lfp}^{\subseteq} F'_P) \subseteq Q'\}\\
& \quad \text{≀}(\subseteq)\ \exists Q\ .\ \mathsf{post}[\![\neg\mathtt{B}]\!](\mathsf{lfp}^{\subseteq} F'_P) \subseteq Q \wedge Q \subseteq Q' \text{ and transitivity;}\\
& \quad (\supseteq) \text{ take } Q = Q'\text{≀}\\
=\ & \{\langle P',\ Q'\rangle \mid \exists P, Q\ .\ P' \subseteq P \wedge \mathsf{lfp}^{\subseteq} F'_P \subseteq Q \wedge \mathsf{post}[\![\neg\mathtt{B}]\!](Q) \subseteq Q'\}\\
& \quad \text{≀}(\subseteq) \text{ take } Q = \mathsf{lfp}^{\subseteq} F'_P; \quad (\supseteq)\ \mathsf{post}[\![\mathtt{B}]\!]P = P \cap \mathcal{B}[\![\mathtt{B}]\!] \text{ so } \mathsf{post}[\![\mathtt{B}]\!] \text{ is increasing≀}\\
=\ & \{\langle P',\ Q'\rangle \mid \exists P, Q, I\ .\ P' \subseteq P \wedge F'_P(I) \subseteq I \wedge I \subseteq Q \wedge \mathsf{post}[\![\neg\mathtt{B}]\!](Q) \subseteq Q'\}\\
& && \text{≀Park fixpoint induction Th. 2≀}\\
=\ & \{\langle P',\ Q'\rangle \mid \exists Q, I\ .\ F'_{P'}(I) \subseteq I \wedge I \subseteq Q \wedge \mathsf{post}[\![\neg\mathtt{B}]\!](Q) \subseteq Q'\}\\
& \quad \text{≀}(\subseteq) \text{ union hence } F'_P(X) = P \cup \mathsf{post}([\![\mathtt{B}]\!] \mathbin{⨾} [\![\mathtt{S}]\!])X \text{ is } \subseteq\text{-increasing in } P \text{ so}\\
& \quad P' \subseteq P \text{ implies } F'_{P'}(I) \subseteq F'_P(I) \subseteq I.\ (\supseteq) \text{ take } P = P'\text{≀}\\
=\ & \{\langle P,\ Q'\rangle \mid \exists I\ .\ F'_P(I) \subseteq I \wedge \mathsf{post}[\![\neg\mathtt{B}]\!](I) \subseteq Q'\}
\end{aligned}$$

$\wr$Rename P' into P. ($\subseteq$) $I \subseteq Q$ implies $\mathsf{post}[\![\neg \mathtt{B}]\!](I) \subseteq \mathsf{post}[\![\neg \mathtt{B}]\!](Q)$ since $\mathsf{post}[\![\neg \mathtt{B}]\!]$ is increasing hence $\mathsf{post}[\![\neg \mathtt{B}]\!](I) \subseteq Q'$ by transitivity; ($\supseteq$) take $Q = I\int$

$= \{\langle P,\, Q\rangle \mid \exists I \,.\, P \cup \mathsf{post}([\![\mathtt{B}]\!] \,⨾\, [\![\mathtt{S}]\!])(I) \subseteq I \wedge \mathsf{post}[\![\neg \mathtt{B}]\!](I) \subseteq Q\}$

$\wr$renaming Q' into Q, def. $F'_P\int$

$= \{\langle P,\, Q\rangle \mid \exists I \,.\, P \subseteq I \wedge \mathsf{post}([\![\mathtt{B}]\!] \,⨾\, [\![\mathtt{S}]\!])I \subseteq I \wedge \mathsf{post}[\![\neg \mathtt{B}]\!](I) \subseteq Q\}$$\wr$def. $\subseteq$ and $\cup\int$

$= \{\langle P,\, Q\rangle \mid \exists I \,.\, P \subseteq I \wedge \mathsf{post}[\![\mathtt{S}]\!](\mathsf{post}[\![\mathtt{B}]\!]I) \subseteq I \wedge \mathsf{post}[\![\neg \mathtt{B}]\!](I) \subseteq Q\}$

$\wr$composition $\mathsf{post}(X \,⨾\, Y) = \mathsf{post}(Y) \circ \mathsf{post}(X)\int$

$= \{\langle P,\, Q\rangle \mid \exists I \,.\, P \subseteq I \wedge \mathsf{post}[\![\mathtt{S}]\!](I \cap \mathcal{B}[\![\mathtt{B}]\!]) \subseteq I \wedge (I \cap \neg\mathcal{B}[\![\mathtt{B}]\!]) \subseteq Q\}$

$\wr\mathsf{post}[\![\mathtt{B}]\!]P = P \cap \mathcal{B}[\![\mathtt{B}]\!]\int$

$= \{\langle P, Q\rangle \mid \exists I \,.\, P \subseteq I \wedge \langle I \cap \mathcal{B}[\![\mathtt{B}]\!], I\rangle \in \{\langle P, Q\rangle \mid \mathsf{post}[\![\mathtt{S}]\!]P \subseteq Q\} \wedge (I \cap \neg\mathcal{B}[\![\mathtt{B}]\!]) \subseteq Q\}$ $\wr$def. $\in\int$

$= \{\langle P, Q\rangle \mid \exists I \,.\, P \subseteq I \wedge \langle I \cap \mathcal{B}[\![\mathtt{B}]\!], I\rangle \in \{\langle P, Q\rangle \mid \exists P', Q' \,.\, P \subseteq P' \wedge \mathsf{post}[\![\mathtt{S}]\!]P' \subseteq Q' \wedge Q' \subseteq Q\} \wedge (I \cap \neg\mathcal{B}[\![\mathtt{B}]\!]) \subseteq Q\}$

$\wr$($\Rightarrow$) Take $P' = P$ and $Q' = Q$. ($\Leftarrow$) $P \subseteq P' \wedge \mathsf{post}[\![\mathtt{S}]\!]P' \subseteq Q' \wedge Q' \subseteq Q$ implies $\mathsf{post}[\![\mathtt{S}]\!]P \subseteq Q$ by transitivity.$\int$

$= \{\langle P,\, Q\rangle \mid \exists I \,.\, P \subseteq I \wedge \langle I \cap \mathcal{B}[\![\mathtt{B}]\!],\, I\rangle \in \mathsf{post}(\supseteq.\subseteq) \circ \mathcal{T}[\![\mathtt{S}]\!] \wedge (I \cap \neg\mathcal{B}[\![\mathtt{B}]\!]) \subseteq Q\}$

$\wr$def. post and $\mathcal{T}[\![\mathtt{S}]\!]\int$

$= \{\langle P,\, Q\rangle \mid \exists I \,.\, P \subseteq I \wedge \langle I \cap \mathcal{B}[\![\mathtt{B}]\!],\, I\rangle \in T_{\mathrm{HL}}(\mathtt{S}) \wedge (I \cap \neg\mathcal{B}[\![\mathtt{B}]\!]) \subseteq Q\}$

$\wr$def. $T_{\mathrm{HL}}\int$ $\square$

Defining $\{\,P\,\}\,\mathtt{S}\,\{\,Q\,\} \triangleq \langle P,\, Q\rangle \in \mathcal{T}[\![\mathtt{S}]\!]$, we can now derive the Hoare rules. For conditional iteration, it is

$$\frac{P \subseteq I,\ \ \{I \cap \mathcal{B}[\![\mathtt{B}]\!]\}\,\mathtt{S}\,\{I\},\ \ (I \cap \neg\mathcal{B}[\![\mathtt{B}]\!]) \subseteq Q}{\{P\}\,\mathtt{while\ (B)\ S}\,\{Q\}} \tag{9}$$

Proof (of (9)). By structural induction (`S` being a strict component of `while (B) S`), the rule sfor $\{P\}\,\mathtt{S}\,\{Q\}$ have already been defined. By Aczel method, the (constant) fixpoint $\mathsf{lfp}^{\subseteq}\, \boldsymbol{\lambda} X \bullet S$ is defined by $\{\frac{\emptyset}{c} \mid c \in S\}$. So for `while (B) S` we have an axiom $\dfrac{\emptyset}{\{P\}\,\mathtt{while\ (B)\ S}\,\{Q\}}$ with side condition $P \subseteq I,\ \{I \cap \mathcal{B}[\![\mathtt{B}]\!]\}\,\mathtt{S}\,\{I\},\ (I \cap \neg\mathcal{B}[\![\mathtt{B}]\!]) \subseteq Q$. Traditionally, the side condition is written as a premiss, to get (9). $\square$

Notice that the proof system is semantically sound and complete by construction (while using e.g. a logic for predicates might cause inexpressivity of the iteration invariant [2,3]). The proof is machine checkable, if not machine checked!

As a last remark on the calculational design of Hoare logic, observe that post is increasing so that (8) is $\mathcal{T}_{\mathrm{HL}}(\mathtt{S}) = \mathsf{post}(=.\subseteq) \circ \mathcal{T}[\![\mathtt{S}]\!]$. This means that the consequence rule $\frac{\{\,P\,\}\,\mathtt{S}\,\{\,Q\,\},\ Q \subseteq Q'}{\{\,P\,\}\,\mathtt{S}\,\{\,Q'\,\}}$ is complete and the unique necessary use of the precondition under approximation is that $P \subseteq I$ of the invariant in the iteration rule (9).

9 Calculational Design of Incorrectness Logic

The incorrectness logic theory (7) is the $\subseteq$-order dual of the Hoare logic theory (6). So the rules for statements, but for iteration, are the same because all correspond to the strongest postcondition logic theory (4) together with consequence rule $\dfrac{P' \subseteq P,\ \{\,P\,\}\,\mathtt{S}\,\{\,Q\,\},\ Q \subseteq Q'}{\{\,P'\,\}\,\mathtt{S}\,\{\,Q'\,\}}$ for Hoare logic and the dual $\dfrac{P' \supseteq P,\ \{\,P\,\}\,\mathtt{S}\,\{\,Q\,\},\ Q \supseteq Q'}{\{\,P'\,\}\,\mathtt{S}\,\{\,Q'\,\}}$ for incorrectness logic. As for iteration, fixpoint induction is the over approximation theorem 2 for Hoare logic and under approximation theorem 3 for incorrectness logic.

For iteration $\mathtt{W} = \mathtt{while\ (B)\ S}$, the theory of incorrectness logic is

$$\begin{aligned}\mathcal{T}_{\mathrm{IL}}[\![\mathtt{W}]\!] &\triangleq \mathsf{post}(\subseteq.\supseteq)(\mathcal{T}[\![\mathtt{W}]\!]) \\ &= \{\langle P,\ Q\rangle \mid \exists\langle J^n,\ n \in \mathbb{N}\rangle\ .\ J^0 = P \wedge \langle J^n \cap \mathcal{B}[\![\mathtt{B}]\!],\ J^{n+1}\rangle \in \mathcal{T}_{\mathrm{IL}}[\![\mathtt{S}]\!] \wedge \\ &\qquad Q \subseteq (\textstyle\bigcup_{n\in\mathbb{N}} J^n) \cap \mathcal{B}[\![\neg\mathtt{B}]\!]\}\end{aligned} \tag{10}$$

(which is similar to OHearn backward variant [17] since the consequence rule can also be separated).

Proof (of (10)). We let $\mathtt{W} = \mathtt{while\ (B)\ S}$.

$\mathcal{T}_{\mathrm{IL}}[\![\mathtt{W}]\!]$

$= \mathsf{post}(\subseteq.\supseteq)(\mathcal{T}[\![\mathtt{W}]\!])$ $\quad$ ⟅def. (7) of $\mathcal{T}_{\mathrm{IL}}$⟆

$= \{\langle P,\ Q\rangle \mid \exists\langle P',\ Q'\rangle \in \mathcal{T}[\![\mathtt{W}]\!]\ .\ \langle P',\ Q'\rangle \subseteq.\supseteq \langle P,\ Q\rangle\}$ $\quad$ ⟅def. post⟆

$= \{\langle P,\ Q\rangle \mid \exists\langle P',\ Q'\rangle \in \mathcal{T}[\![\mathtt{W}]\!]\ .\ P' \subseteq P \wedge Q \subseteq Q'\}$ $\quad$ ⟅def. $\subseteq.\supseteq$⟆

$= \{\langle P,\ Q\rangle \mid \exists\langle P',\ Q'\rangle \in \{\langle P'',\ \mathsf{post}[\![\mathtt{W}]\!]P''\rangle \mid P'' \in \wp(\Sigma)\}\ .\ P' \subseteq P \wedge Q \subseteq Q'\}$
⟅def. (4) of $\mathcal{T}[\![\mathtt{W}]\!]$⟆

$= \{\langle P,\ Q\rangle \mid \exists P', Q', P''\ .\ P' = P'' \wedge Q' = \mathsf{post}[\![\mathtt{W}]\!]P'' \wedge P' \subseteq P \wedge Q \subseteq Q'\}$
⟅def. $\in$⟆

$= \{\langle P,\ Q\rangle \mid \exists P'\ .\ P' \subseteq P \wedge Q \subseteq \mathsf{post}[\![\mathtt{W}]\!]P'\}$ $\quad$ ⟅def. $=$⟆

$= \{\langle P,\ Q\rangle \mid Q \subseteq \mathsf{post}[\![\mathtt{W}]\!]P\}$
⟅($\subseteq$) $\mathsf{post}[\![\mathtt{W}]\!]$ increasing and transitivity; ($\supseteq$) take $P' = P$ and reflexivity⟆

$= \{\langle P,\ Q\rangle \mid Q \subseteq \mathsf{post}[\![\neg\mathtt{B}]\!](\mathsf{lfp}^{\subseteq} F'_P)\}$ $\quad$ ⟅(5) with $F'_P(X) \triangleq P \cup \mathsf{post}([\![\mathtt{B}]\!] \mathbin{;} [\![\mathtt{S}]\!])X$⟆

$= \{\langle P,\ Q\rangle \mid \exists I\ .\ Q \subseteq \mathsf{post}[\![\neg\mathtt{B}]\!](I) \wedge I \subseteq \mathsf{lfp}^{\subseteq} F'_P\}$
⟅($\subseteq$) Take $I = \mathsf{lfp}^{\subseteq} F'_P$ and reflexivity;
($\supseteq$) By Galois connection $\langle\wp(\mathcal{X}),\ \subseteq\rangle \underset{\mathsf{post}(r)}{\overset{\widetilde{\mathsf{pre}}(r)}{\leftrightarrows}} \langle\wp(\mathcal{Y}),\ \subseteq\rangle$, $\mathsf{post}[\![\neg\mathtt{B}]\!]$ is increasing so $Q \subseteq \mathsf{post}[\![\neg\mathtt{B}]\!](I) \subseteq \mathsf{post}[\![\neg\mathtt{B}]\!](\mathsf{lfp}^{\subseteq} F'_P)$ and transitivity⟆

$= \{\langle P,\ Q\rangle \mid \exists I\ .\ Q \subseteq \mathsf{post}[\![\neg\mathtt{B}]\!](I) \wedge \exists\langle J^n, n < \omega\rangle\ .\ J^0 = \emptyset \wedge J^{n+1} \subseteq F'_P(J^n) \wedge I \subseteq \bigcup_{n<\omega} J^n\}$ $\quad$ ⟅fixpoint under approximation Th. II.3.6⟆

$= \{\langle P,\, Q\rangle \mid \exists\langle J^n,\, n<\omega\rangle \,.\, J^0=\emptyset \wedge J^{n+1}\subseteq F'_P(J^n) \wedge Q \subseteq \mathsf{post}[\![\neg\mathtt{B}]\!](\bigcup_{n<\omega} J^n)\}$

$\wr(\subseteq)$ By Galois connection $\langle\wp(\mathcal{X}),\, \subseteq\rangle \xrightleftharpoons[\mathsf{post}(r)]{\widetilde{\mathsf{pre}}(r)} \langle\wp(\mathcal{Y}),\, \subseteq\rangle$ $\mathsf{post}[\![\neg\mathtt{B}]\!]$ is increasing so $Q\subseteq \mathsf{post}[\![\neg\mathtt{B}]\!](I) \subseteq \mathsf{post}[\![\neg\mathtt{B}]\!](\bigcup_{n<\omega} J^n)$ and transitivity; $(\supseteq)$ take $I=\bigcup_{n<\omega} J^n\wr$

$= \{\langle P,\, Q\rangle \mid \exists\langle J^n,\, n<\omega\rangle \,.\, J^0=\emptyset \wedge J^{n+1}\subseteq (P\cup \mathsf{post}([\![\mathtt{B}]\!]\,{}_{9}^{\circ}\,[\![\mathtt{S}]\!])(J^n)) \wedge Q \subseteq \mathsf{post}[\![\neg\mathtt{B}]\!](\bigcup_{n<\omega} J^n)\}$ $\wr$def. $F'_P\wr$

$= \{\langle P,\, Q\rangle \mid \exists\langle J^n,\, 1\leqslant n<\omega\rangle \,.\, J^1=P \wedge J^{n+1}\subseteq \mathsf{post}([\![\mathtt{B}]\!]\,{}_{9}^{\circ}\,[\![\mathtt{S}]\!])(J^n) \wedge Q \subseteq \mathsf{post}[\![\neg\mathtt{B}]\!](\bigcup_{1\leqslant n<\omega} J^n)\}$ $\wr$getting rid of $J^0=\emptyset\wr$

$= \{\langle P,\, Q\rangle \mid \exists\langle J^n,\, n\in\mathbb{N}\rangle \,.\, J^0=P \wedge J^{n+1}\subseteq \mathsf{post}([\![\mathtt{B}]\!]\,{}_{9}^{\circ}\,[\![\mathtt{S}]\!])(J^n) \wedge Q \subseteq \mathsf{post}[\![\neg\mathtt{B}]\!](\bigcup_{n\in\mathbb{N}} J^n)\}$ $\wr$changing $n+1$ to $n\wr$

$= \{\langle P,\, Q\rangle \mid \exists\langle J^n,\, n\in\mathbb{N}\rangle \,.\, J^0=P \wedge J^{n+1}\subseteq \mathsf{post}[\![\mathtt{S}]\!](J^n\cap\mathcal{B}[\![\mathtt{B}]\!]) \wedge Q \subseteq (\bigcup_{n\in\mathbb{N}} J^n)\cap\mathcal{B}[\![\neg\mathtt{B}]\!]\}$ $\wr\mathsf{post}[\![\mathtt{B}]\!]P = P\cap\mathcal{B}[\![\mathtt{B}]\!]\wr$

$= \{\langle P,\, Q\rangle \mid \exists\langle J^n,\, n\in\mathbb{N}\rangle \,.\, J^0=P \wedge \langle J^n\cap\mathcal{B}[\![\mathtt{B}]\!],\, J^{n+1}\rangle \in \{\langle P',\, Q'\rangle \mid Q'\subseteq \mathsf{post}[\![\mathtt{S}]\!])P)\} \wedge Q \subseteq (\bigcup_{n\in\mathbb{N}} J^n)\cap\mathcal{B}[\![\neg\mathtt{B}]\!]\}$ $\wr$def. $\in\wr$

$= \{\langle P,\, Q\rangle \mid \exists\langle J^n,\, n\in\mathbb{N}\rangle \,.\, J^0=P \wedge \langle J^n\cap\mathcal{B}[\![\mathtt{B}]\!],\, J^{n+1}\rangle \in \mathcal{T}_{\mathrm{IL}}[\![\mathtt{S}]\!] \wedge Q \subseteq (\bigcup_{n\in\mathbb{N}} J^n)\cap\mathcal{B}[\![\neg\mathtt{B}]\!]\}$ $\wr$def. $\mathcal{T}_{\mathrm{IL}}\wr$ □

Defining $[\,P\,]\,\mathtt{S}\,[\,Q\,] \triangleq \langle P,\, Q\rangle \in \mathcal{T}_{\mathrm{IL}}[\![\mathtt{S}]\!]$, the calculational design of incorrectness logic rules is as follows

$$\frac{J^0=P,\ [J^n\cap\mathcal{B}[\![\mathtt{B}]\!]]\,\mathtt{S}\,[J^{n+1}],\ Q\subseteq(\bigcup_{n\in\mathbb{N}} J^n)\cap\mathcal{B}[\![\neg\mathtt{B}]\!]}{[P]\,\mathtt{while\ (B)\ S}\,[Q]} \tag{11}$$

Proof (of (11)). By structural induction (S being a strict component of `while (B) S`), the rule for $[P]\,\mathtt{S}\,[Q]$ have already been defined. By Aczel method, the (constant) fixpoint $\mathsf{lfp}^{\subseteq}\boldsymbol{\lambda} X\bullet S$ is defined by $\{\frac{\emptyset}{c} \mid c\in S\}$. So for `while (B) S` we have an axiom $\frac{\emptyset}{\{P\}\,\mathtt{while\ (B)\ S}\,\{Q\}}$ with side condition $J^0=P,\ [J^n\cap\mathcal{B}[\![\mathtt{B}]\!]]\,\mathtt{S}\,[J^{n+1}],\ Q\subseteq(\bigcup_{n\in\mathbb{N}} J^n)\cap\mathcal{B}[\![\neg\mathtt{B}]\!]$; Traditionally, the side condition is written as a premiss, to get (11). □

10 On Hoare Incorrectness

Incorrectness logic, a variant of reverse Hoare logic [21] was introduced by [17] as a counterpoint of Hoare logic *"When reasoning informally about a program, people make abstract inferences about what might go wrong, as well as about what* *must go right. [...] We explore our hypothesis by defining incorrectness logic, a formalism that is similar to Hoare's logic of program correctness [13], except that it is oriented to proving incorrectness rather than correctness."* However, incorrectness logic is not Hoare incorrectness logic. It is sufficient but not necessary. Assuming $Q \neq \Sigma$, we have

$$\begin{aligned} \neg(\{P\}\,\mathtt{S}\,\{Q\}) &\overset{\not\Rightarrow}{\Leftarrow} [\,P\,]\,\mathtt{S}\,[\,\neg Q\,] \\ &\Leftrightarrow \exists R \in \wp(\Sigma)\ .\ [\,P\,]\,\mathtt{S}\,[\,R\,] \wedge R \cap \neg Q \neq \emptyset \\ &\Leftrightarrow \exists \sigma \in \Sigma\ .\ [\,P\,]\,\mathtt{S}\,[\,\{\sigma\}\,] \wedge \sigma \notin Q \end{aligned} \tag{12}$$

The incompleteness of incorrectness logic $[\,P\,]\,\mathtt{S}\,[\,\neg Q\,]$ to prove Hoare incorrectness $\neg(\{\,P\,\}\,\mathtt{S}\{\,Q\,\})$ comes from the fact that we may have $\{\,P\,\}\,\mathtt{S}\{\,Q'\,\}$ for some $Q' \subseteq Q$. Therefore we have to select $R = \neg Q \setminus Q'$ to ensure completeness. However, the formula $\exists R \in \wp(\Sigma)\ .\ [\,P\,]\,\mathtt{S}\,[\,R\,] \wedge R \cap \neg Q \neq \emptyset$ is not a formula of the incorrectness logic. Moreover, R can be reduced to a single state $\{\sigma\}$ since a single counter example is sufficient to prove Hoare incorrectness. This leads to Hoare incorrectness logic.

Proof (of (12)).

– $[\,P\,]\,\mathtt{S}\,[\,\neg Q\,]$

$\Leftrightarrow \langle P,\ \neg Q\rangle \in \mathcal{T}_{\mathrm{IL}}[\![\mathtt{S}]\!]$ $\quad$ ⦃incorrectness triple definition⦄

$\Leftrightarrow \neg Q \subseteq \mathsf{post}[\![\mathtt{S}]\!]P$ $\quad$ ⦃def. (7) of $\mathcal{T}_{\mathrm{IL}}[\![\mathtt{S}]\!]$⦄

$\Rightarrow \mathsf{post}[\![\mathtt{S}]\!]P \cap (\neg Q) \neq \emptyset$ $\quad$ ⦃assuming $Q \neq \Sigma$ so $(\neg Q) \neq \emptyset$⦄

$\Leftrightarrow \neg(\mathsf{post}[\![\mathtt{S}]\!]P \subseteq Q)$ $\quad$ ⦃def. $\subseteq$⦄

$\Leftrightarrow \neg(\langle P,\ Q\rangle \in \{\langle P,\ Q\rangle \mid \mathsf{post}[\![\mathtt{S}]\!]P \subseteq Q\})$ $\quad$ ⦃def. $\in$⦄

$\Leftrightarrow \neg(\langle P,\ Q\rangle \in \mathcal{T}_{\mathrm{HL}}[\![\mathtt{S}]\!])$ $\quad$ ⦃def. (6) of $\mathcal{T}_{\mathrm{HL}}[\![\mathtt{S}]\!]$⦄

$\Leftrightarrow \neg(\{P\}\,\mathtt{S}\{Q\})$ $\quad$ ⦃Hoare triple definition, Q.E.D.⦄

The converse is not true, as shown by the counter example $\neg(\{\mathsf{true}\}\ \mathtt{x = 0}\{\mathtt{x} \neq 0 \wedge \mathtt{x} \neq 1\})$ holds but not $[\mathsf{true}]\ \mathtt{x = 0}[\mathtt{x} = 0 \vee \mathtt{x} = 1]$.

– $\neg(\{P\}\,\mathtt{S}\{Q\})$ $\quad$ ⦃def. incorrect Hoare triple⦄

$\Leftrightarrow \neg(\langle P,\ Q\rangle \in \mathcal{T}_{\mathrm{HL}}[\![\mathtt{S}]\!])$ $\quad$ ⦃def. Hoare triple⦄

$\Leftrightarrow \neg(\langle P,\ Q\rangle \in \{\langle P,\ Q\rangle \mid \mathsf{post}[\![\mathtt{S}]\!]P \subseteq Q\})$ $\quad$ ⦃def. (6) of $\mathcal{T}_{\mathrm{HL}}[\![\mathtt{S}]\!]$⦄

$\Leftrightarrow \neg(\mathsf{post}[\![\mathtt{S}]\!]P \subseteq Q)$ $\quad$ ⦃def. $\in$⦄

$\Leftrightarrow \neg(\{\sigma' \mid \exists \sigma \in P\ .\ \langle \sigma,\ \sigma'\rangle \in [\![\mathtt{S}]\!]\} \subseteq Q)$ $\quad$ ⦃def. post⦄

$\Leftrightarrow \neg(\forall \sigma'\ .\ (\exists \sigma \in P\ .\ \langle \sigma,\ \sigma'\rangle \in [\![\mathtt{S}]\!]) \Rightarrow (\sigma' \in Q))$ $\quad$ ⦃def. $\subseteq$⦄

$\Leftrightarrow \exists \sigma'\ .\ \exists \sigma \in P\ .\ \langle \sigma,\ \sigma'\rangle \in [\![\mathtt{S}]\!] \wedge \sigma' \notin Q$ $\quad$ ⦃def. negation $\neg$⦄

$\Leftrightarrow \exists \sigma \notin Q\ .\ \exists \sigma' \in P\ .\ \langle \sigma',\ \sigma\rangle \in [\![\mathtt{S}]\!]$ $\quad$ ⦃commutativity and renaming⦄

$\Leftrightarrow \exists \sigma \in \Sigma\ .\ \exists \sigma' \in P\ .\ \langle \sigma',\ \sigma\rangle \in [\![\mathtt{S}]\!] \wedge \sigma \notin Q$ $\quad$ ⦃def. $\exists$⦄

$\Leftrightarrow \exists \sigma \in \Sigma\ .\ \forall \sigma'' \in \{\sigma\}\ .\ \exists \sigma' \in P\ .\ \langle \sigma',\ \sigma''\rangle \in [\![\mathtt{S}]\!] \wedge \sigma \notin Q$ $\quad$ ⦃def. $\in$⦄

$\Leftrightarrow \exists \sigma \in \Sigma\ .\ \{\sigma\} \subseteq \{\sigma'' \mid \exists \sigma' \in P\ .\ \langle \sigma',\ \sigma''\rangle \in [\![\mathtt{S}]\!]\} \wedge \sigma \notin Q$ $\quad$ ⦃def. $\subseteq$⦄

$\Leftrightarrow \exists \sigma \in \Sigma \,.\, \{\sigma\} \subseteq \mathsf{post}[\![\mathtt{S}]\!]P\} \wedge \sigma \notin Q$ $\qquad \wr$def. $\mathsf{post}[\![\mathtt{S}]\!]$$\int$

$\Leftrightarrow \exists \sigma \in \Sigma \,.\, \langle P,\ \{\sigma\}\rangle \in \{\langle P,\ Q\rangle \mid Q \subseteq \mathsf{post}[\![\mathtt{S}]\!]P\} \wedge \sigma \notin Q$ $\qquad \wr$def. $\in$$\int$

$\Leftrightarrow \exists \sigma \in \Sigma \,.\, \langle P,\ \{\sigma\}\rangle \in \mathcal{T}_{\mathrm{IL}}[\![\mathtt{S}]\!] \wedge \sigma \notin Q$ $\qquad \wr$def. (7) of $\mathcal{T}_{\mathrm{IL}}[\![\mathtt{S}]\!]$$\int$

$\Leftrightarrow \exists \sigma \in \Sigma \,.\, [P]\,\mathtt{S}\,[\{\sigma\}] \wedge \sigma \notin Q$ $\qquad \wr$def. incorrectness logic triple, Q.E.D.$\int$

$\Leftrightarrow \exists R \in \wp(\Sigma) \,.\, [P]\,\mathtt{S}\,[R] \wedge R \cap \neg Q \neq \emptyset$

$\wr$($\subseteq$) take $R = \{\sigma\}$;
($\supseteq$) since $R \cap \neg Q \neq \emptyset$, we have $\exists \sigma \in R \,.\, \sigma \notin Q$ and $[P]\,\mathtt{S}\,[\{\sigma\}]$ since otherwise we would have $\neg(\forall \sigma'' \in \{\sigma\} \,.\, \exists \sigma' \in P \,.\, \langle \sigma'',\ \sigma'\rangle \in [\![\mathtt{S}]\!]) \Leftrightarrow \forall \sigma' \in P \,.\, \langle \sigma,\ \sigma'\rangle \notin [\![\mathtt{S}]\!])$, in contradiction with $[P]\,\mathtt{S}\,[R]$ and $\sigma \in R$. $\int$ $\square$

11 Calculational Design of Hoare Incorrectness Logic

Incorrectness logic being incomplete to prove Hoare incorrectness, we design a sound and semantically complete Hoare incorrectness logic $\overline{\mathrm{HL}}$.

Let us consider the negation of $X \in \wp(\mathcal{X})$, to be $\alpha^{\neg}(X) \triangleq \neg X$ (where $\neg X \triangleq \mathcal{X} \setminus X$) with Galois isomorphisms $\langle \wp(\mathcal{X}),\ \subseteq\rangle \underset{\alpha^{\neg}}{\overset{\alpha^{\neg}}{\leftrightarrows}} \langle \wp(\mathcal{X}),\ \supseteq\rangle$ and $\langle \wp(\mathcal{X}),\ \supseteq\rangle \underset{\alpha^{\neg}}{\overset{\alpha^{\neg}}{\leftrightarrows}} \langle \wp(\mathcal{X}),\ \subseteq\rangle$. The theory of Hoare incorrectness logic is

$$\mathcal{T}_{\overline{\mathrm{HL}}}[\![\mathtt{S}]\!] \triangleq \alpha^{\neg}(\mathcal{T}_{\mathrm{HL}}[\![\mathtt{S}]\!]) \tag{13}$$

For iteration $\mathtt{W} = \mathtt{while\ (B)\ S}$, the theory of Hoare incorrectness logic is

$$\mathcal{T}_{\overline{\mathrm{HL}}}[\![\mathtt{W}]\!] = \{\langle P,\ Q\rangle \mid \exists n \geqslant 1 \,.\, \exists \langle \sigma_i \in I,\ i \in [1, n]\rangle \,.\, \sigma_1 \in P \wedge \forall i \in [1, n[\,.\, \langle \mathcal{B}[\![\mathtt{B}]\!] \cap \{\sigma_i\},\ \neg\{\sigma_{i+1}\}\rangle \in \mathcal{T}_{\overline{\mathrm{HL}}}[\![\mathtt{S}]\!] \wedge \sigma_n \notin \mathcal{B}[\![\mathtt{B}]\!] \wedge \sigma_n \notin Q\} \tag{14}$$

Proof (of (14)).

$\mathcal{T}_{\overline{\mathrm{HL}}}[\![\mathtt{W}]\!]$

$= \alpha^{\neg}(\mathcal{T}_{\mathrm{HL}}[\![\mathtt{W}]\!])$ $\qquad \wr$def. (13) of $\mathcal{T}_{\overline{\mathrm{HL}}}[\![\mathtt{W}]\!]$$\int$

$= \alpha^{\neg}(\{\langle P,\ Q\rangle \mid \mathsf{post}[\![\mathtt{W}]\!]P \subseteq Q\})$ $\qquad \wr$def. (6) of $\mathcal{T}_{\mathrm{HL}}[\![\mathtt{W}]\!]$$\int$

$= \{\langle P,\ Q\rangle \mid \neg(\mathsf{post}[\![\mathtt{W}]\!]P \subseteq Q)\}$ $\qquad \wr$def. $\alpha^{\neg}$$\int$

$= \{\langle P,\ Q\rangle \mid \mathsf{post}[\![\mathtt{W}]\!]P \cap \neg Q \neq \emptyset\}$ $\qquad \wr$def. $\subseteq$ and $\neg$$\int$

$= \{\langle P,\ Q\rangle \mid \mathsf{post}[\![\neg\mathtt{B}]\!](\mathsf{lfp}^{\subseteq} F'_P) \cap \neg Q \neq \emptyset\}$ $\wr$(5), $F'_P(X) \triangleq P \cup \mathsf{post}([\![\mathtt{B}]\!] \fatsemi [\![\mathtt{S}]\!])X$$\int$

$= \{\langle P,\ Q\rangle \mid \mathsf{lfp}^{\subseteq} F'_P \cap \mathsf{pre}[\![\neg\mathtt{B}]\!](\neg Q) \neq \emptyset\}$
$\qquad \wr$$\mathsf{post}(R)P \cap Q \neq \emptyset \Leftrightarrow P \cap \mathsf{pre}(R)Q \neq \emptyset$$\int$

$= \{\langle P,\ Q\rangle \mid \exists I \in \wp(\Sigma) \,.\, F'_P(I) \subseteq I \wedge \exists \langle W,\ \leqslant\rangle \in \mathfrak{Wf} \,.\, \exists \nu \in I \to W \,.\, \exists \langle \sigma_i \in I,\ i \in [1, \infty]\rangle \,.\, \sigma_1 \in F'_P(\emptyset) \wedge \forall i \in [1, \infty] \,.\, \sigma_{i+1} \in F'_P(\{\sigma_i\}) \wedge \forall i \in [1, \infty] \,.\, (\sigma_i \neq \sigma_{i+1}) \Rightarrow (\nu(\sigma_i) > \nu(\sigma_{i+1}) \wedge \forall i \in [1, \infty] \,.\, (\nu(\sigma_i) \not> \nu(\sigma_{i+1}) \Rightarrow \{\sigma_i\} \cap \mathsf{pre}[\![\neg\mathtt{B}]\!](\neg Q) \neq 0\}$ $\qquad \wr$induction principle Th. 4$\int$

$= \{\langle P,\ Q\rangle \mid \exists I \in \wp(\Sigma) \,.\, P \subseteq I \wedge \mathsf{post}([\![\mathtt{B}]\!] \fatsemi [\![\mathtt{S}]\!])I \subseteq I \wedge \exists \langle W,\ \leqslant\rangle \in \mathfrak{Wf} \,.\, \exists \nu \in I \to W \,.\, \exists \langle \sigma_i \in I,\ i \in [1, \infty]\rangle \,.\, \sigma_1 \in P \wedge \forall i \in [1, \infty] \,.\, (\sigma_{i+1} \in P \vee \{\sigma_{i+1}\} \subseteq \mathsf{post}([\![\mathtt{B}]\!] \fatsemi [\![\mathtt{S}]\!])\{\sigma_i\}) \wedge \forall i \in [1, \infty] \,.\, (\sigma_i \neq \sigma_{i+1}) \Rightarrow (\nu(\sigma_i) > \nu(\sigma_{i+1}) \wedge \forall i \in [1, \infty] \,.\, (\nu(\sigma_i) \not> \nu(\sigma_{i+1}) \Rightarrow \sigma_i \in \mathsf{pre}[\![\neg\mathtt{B}]\!](\neg Q)\}$

$\wr$def. $F'_P(X) \triangleq P \cup \mathsf{post}(\llbracket \mathtt{B} \rrbracket \,⨟\, \llbracket \mathtt{S} \rrbracket)X$, $\subseteq$, and post, which is $\emptyset$-strict$\int$

$= \{\langle P,\ Q\rangle \mid \exists I \in \wp(\Sigma) \ .\ P \subseteq I \wedge \mathsf{post}(\llbracket \mathtt{B} \rrbracket \,⨟\, \llbracket \mathtt{S} \rrbracket)I \subseteq I \wedge \exists \langle W,\ \leqslant\rangle \in \mathfrak{Wf}\ .\ \exists \nu \in I \to W\ .\ \exists \langle \sigma_i \in I,\ i \in [1,\infty]\rangle\ .\ \sigma_1 \in P \wedge \forall i \in [1,\infty]\ .\ \{\sigma_{i+1}\} \subseteq \mathsf{post}(\llbracket \mathtt{B} \rrbracket \,⨟\, \llbracket \mathtt{S} \rrbracket)\{\sigma_i\} \wedge \forall i \in [1,\infty]\ .\ (\sigma_i \neq \sigma_{i+1}) \Rightarrow (\nu(\sigma_i) > \nu(\sigma_{i+1}) \wedge \forall i \in [1,\infty]\ .\ (\nu(\sigma_i) \not> \nu(\sigma_{i+1}) \Rightarrow \sigma_i \in \mathsf{pre}\llbracket \neg\mathtt{B} \rrbracket(\neg Q)\}$

$\wr$since if $\sigma_{i+1} \in P$, we can equivalently consider the sequence $\langle \sigma_j \in I,\ j \in [i+1,\infty]\rangle$$\int$

$= \{\langle P,\ Q\rangle \mid \exists I \in \wp(\Sigma)\ .\ P \subseteq I \wedge \mathsf{post}(\llbracket \mathtt{B} \rrbracket \,⨟\, \llbracket \mathtt{S} \rrbracket)I \subseteq I \wedge \exists n \geqslant 1\ .\ \exists \langle \sigma_i \in I,\ i \in [1,n]\rangle\ .\ \sigma_1 \in P \wedge \forall i \in [1,n[\ .\ \{\sigma_{i+1}\} \subseteq \mathsf{post}(\llbracket \mathtt{B} \rrbracket \,⨟\, \llbracket \mathtt{S} \rrbracket)\{\sigma_i\} \wedge \sigma_n \in \mathsf{pre}\llbracket \neg\mathtt{B} \rrbracket(\neg Q)\}$

$\wr$($\subseteq$) By $\langle W,\ \leqslant\rangle \in \mathfrak{Wf}$, $\nu \in I \to W$, $\forall i \in [1,\infty]\ .\ (\sigma_i \neq \sigma_{i+1}) \Rightarrow (\nu(\sigma_i) > \nu(\sigma_{i+1})$, the sequence is ultimately stationary at some rank n. For then on, $\sigma_{i+1} = \sigma_i$, $i \geqslant n$ and so $\nu(\sigma_i) = \nu(\sigma_{i+1})$. Therefore $\forall i \in [1,\infty]\ .\ (\nu(\sigma_i) \not> \nu(\sigma_{i+1}) \Rightarrow \sigma_i \notin Q$ implies that $\sigma_n \in \mathsf{pre}\llbracket \neg\mathtt{B} \rrbracket(\neg Q)$;

($\supseteq$) Conversely, from $\langle \sigma_i \in I,\ i \in [1,n]\rangle$ we can define $W = \{\sigma_i \mid i \in [1,n]\} \cup \{-\infty\}$ with $-\infty < \sigma_i < \sigma_{i+1}$ and $\nu(x) = \llparenthesis x \in \{\sigma_i \mid i \in [1,n]\ ?\ x : -\infty \rrparenthesis$ and the sequence $\langle \sigma_j \in I,\ j \in [1,\infty]\rangle$ repeats σ_n ad infimum for $j \geqslant n$.$\int$

$= \{\langle P,\ Q\rangle \mid \exists I \in \wp(\Sigma)\ .\ P \subseteq I \wedge \mathsf{post}(\llbracket \mathtt{B} \rrbracket \,⨟\, \llbracket \mathtt{S} \rrbracket)I \subseteq I \wedge \exists n \geqslant 1\ .\ \exists \langle \sigma_i \in I,\ i \in [1,n]\rangle\ .\ \sigma_1 \in P \wedge \forall i \in [1,n[\ .\ \{\sigma_{i+1}\} \subseteq \mathsf{post}(\llbracket \mathtt{B} \rrbracket \,⨟\, \llbracket \mathtt{S} \rrbracket)\{\sigma_i\} \wedge \sigma_n \notin \mathcal{B}\llbracket \mathtt{B} \rrbracket \wedge \sigma_n \notin Q\}$

$\wr$def. $\mathsf{pre}$$\int$

$= \{\langle P,\ Q\rangle \mid \exists n \geqslant 1\ .\ \exists \langle \sigma_i \in I,\ i \in [1,n]\rangle\ .\ \sigma_1 \in P \wedge \forall i \in [1,n[\ .\ \{\sigma_{i+1}\} \subseteq \mathsf{post}(\llbracket \mathtt{B} \rrbracket \,⨟\, \llbracket \mathtt{S} \rrbracket)\{\sigma_i\} \wedge \sigma_n \notin \mathcal{B}\llbracket \mathtt{B} \rrbracket \wedge \sigma_n \notin Q\}$

$\wr$$I$ is not used and can always be chosen to be $\Sigma$$\int$

$= \{\langle P,\ Q\rangle \mid \exists n \geqslant 1\ .\ \exists \langle \sigma_i \in I,\ i \in [1,n]\rangle\ .\ \sigma_1 \in P \wedge \forall i \in [1,n[\ .\ \mathsf{post}(\llbracket \mathtt{B} \rrbracket \,⨟\, \llbracket \mathtt{S} \rrbracket)\{\sigma_i\} \cap \{\sigma_{i+1}\} \neq \emptyset \wedge \sigma_n \notin \mathcal{B}\llbracket \mathtt{B} \rrbracket \wedge \sigma_n \notin Q\}$ $\wr$since $x \in X \Leftrightarrow X \cap \{x\} \neq \emptyset$$\int$

$= \{\langle P,\ Q\rangle \mid \exists n \geqslant 1\ .\ \exists \langle \sigma_i \in I,\ i \in [1,n]\rangle\ .\ \sigma_1 \in P \wedge \forall i \in [1,n[\ .\ \mathsf{post}(\llbracket \mathtt{B} \rrbracket \,⨟\, \llbracket \mathtt{S} \rrbracket)\{\sigma_i\} \cap \neg(\neg\{\sigma_{i+1}\}) \neq \emptyset \wedge \sigma_n \notin \mathcal{B}\llbracket \mathtt{B} \rrbracket \wedge \sigma_n \notin Q\}$ $\wr$def. $\neg X = \Sigma \setminus X$$\int$

$= \{\langle P,\ Q\rangle \mid \exists n \geqslant 1\ .\ \exists \langle \sigma_i \in I,\ i \in [1,n]\rangle\ .\ \sigma_1 \in P \wedge \forall i \in [1,n[\ .\ \neg(\mathsf{post}(\llbracket \mathtt{B} \rrbracket \,⨟\, \llbracket \mathtt{S} \rrbracket)\{\sigma_i\} \subseteq (\neg\{\sigma_{i+1}\})) \wedge \sigma_n \notin \mathcal{B}\llbracket \mathtt{B} \rrbracket \wedge \sigma_n \notin Q\}$ $\wr$$\neg(X \subseteq Y) \Leftrightarrow (X \cap \neg Y \neq \emptyset$$\int$

$= \{\langle P,\ Q\rangle \mid \exists n \geqslant 1\ .\ \exists \langle \sigma_i \in I,\ i \in [1,n]\rangle\ .\ \sigma_1 \in P \wedge \forall i \in [1,n[\ .\ \neg(\mathsf{post}(\llbracket \mathtt{S} \rrbracket)(\mathcal{B}\llbracket \mathtt{B} \rrbracket \cap \{\sigma_i\}) \subseteq (\neg\{\sigma_{i+1}\})) \wedge \sigma_n \notin \mathcal{B}\llbracket \mathtt{B} \rrbracket \wedge \sigma_n \notin Q\}$

$\wr$def. post, $\llbracket \mathtt{B} \rrbracket$, and $⨟$$\int$

$= \{\langle P,\ Q\rangle \mid \exists n \geqslant 1\ .\ \exists \langle \sigma_i \in I,\ i \in [1,n]\rangle\ .\ \sigma_1 \in P \wedge \forall i \in [1,n[\ .\ \langle \mathcal{B}\llbracket \mathtt{B} \rrbracket \cap \{\sigma_i\},\ \neg\{\sigma_{i+1}\}\rangle \in \{\langle P,\ Q\rangle \mid \neg(\mathsf{post}(\llbracket \mathtt{S} \rrbracket)P \subseteq Q)\} \wedge \sigma_n \notin \mathcal{B}\llbracket \mathtt{B} \rrbracket \wedge \sigma_n \notin Q\}$ $\wr$def. $\in$$\int$

$= \{\langle P,\ Q\rangle \mid \exists n \geqslant 1\ .\ \exists \langle \sigma_i \in I,\ i \in [1,n]\rangle\ .\ \sigma_1 \in P \wedge \forall i \in [1,n[\ .\ \langle \mathcal{B}\llbracket \mathtt{B} \rrbracket \cap \{\sigma_i\},\ \neg\{\sigma_{i+1}\}\rangle \in \mathcal{T}_{\overline{\mathrm{HL}}}\llbracket \mathtt{S} \rrbracket \wedge \sigma_n \notin \mathcal{B}\llbracket \mathtt{B} \rrbracket \wedge \sigma_n \in Q\}$ $\wr$def. $\mathcal{T}_{\overline{\mathrm{HL}}}\llbracket \mathtt{S} \rrbracket$$\int$ $\square$

Defining $\llparenthesis P \rrparenthesis\, \mathtt{S}\, \llparenthesis Q \rrparenthesis \triangleq \langle P,\ Q\rangle \in \mathcal{T}_{\overline{\mathrm{HL}}}\llbracket \mathtt{S} \rrbracket$, the calculational design of incorrectness logic rules is as follows,

$$\frac{\begin{array}{l}\exists \langle \sigma_i \in I,\ i \in [1,n]\rangle\ .\ \sigma_1 \in P \wedge \\ \qquad\qquad \forall i \in [1,n[\ .\ \llparenthesis \mathcal{B}\llbracket \mathtt{B} \rrbracket \cap \{\sigma_i\} \rrparenthesis\, \mathtt{S}\, \llparenthesis \neg\{\sigma_{i+1}\} \rrparenthesis \wedge \\ \qquad\qquad \sigma_n \notin \mathcal{B}\llbracket \mathtt{B} \rrbracket \wedge \sigma_n \notin Q\end{array}}{\llparenthesis P \rrparenthesis\, \mathtt{while\ (B)\ S}\, \llparenthesis Q \rrparenthesis} \tag{15}$$

Proof (of (15)).

By structural induction (`S` being a strict component of `while (B) S`), the rule for $(\!|\, P \,|\!)\, \mathtt{S}\, (\!|\, Q \,|\!)$ have already been defined. By Aczel method, the (constant) fixpoint $\mathsf{lfp}^{\subseteq} \boldsymbol{\lambda} X \bullet S$ is defined by $\{\frac{\emptyset}{c} \mid c \in S\}$. So for `while (B) S` we have an axiom $\dfrac{\emptyset}{(\!|\, P \,|\!)\, \mathtt{while\ (B)\ S}\, (\!|\, Q \,|\!)}$ with side condition $\exists \langle \sigma_i \in I,\ i \in [1, n] \rangle$. $\sigma_1 \in P \wedge \forall i \in [1, n[\ .\ (\!|\, \mathcal{B}[\![\mathtt{B}]\!] \cap \{\sigma_i\} \,|\!)\, \mathtt{S}\, (\!|\, \neg\{\sigma_{i+1}\} \,|\!) \wedge \sigma_n \notin \mathcal{B}[\![\mathtt{B}]\!] \wedge \sigma_n \notin Q$ where $(\!|\, \mathcal{B}[\![\mathtt{B}]\!] \cap \{\sigma_i\} \,|\!)\, \mathtt{S}\, (\!|\, \neg\{\sigma_{i+1}\} \,|\!)$ is well-defined by structural induction. Traditionally, the side condition is written as a premiss, to get (15). □

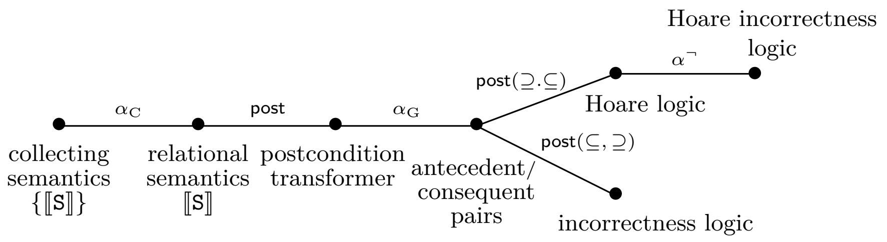
Rule (15) states that $\langle \sigma_i \in I, i \in [1, n] \rangle$ is a finite iteration in the loop starting with P true and finishing with Q false, which is obviously a counter example to Hoare triple $\{P\}\, \mathtt{while\ \ (B)\ \ S}\, \{Q\}$. Notice that, by structural induction, $(\!|\, \mathcal{B}[\![\mathtt{B}]\!] \cap \{\sigma_i\} \,|\!)\, \mathtt{S}\, (\!|\, \neg\{\sigma_{i+1}\} \,|\!)$ enforces the execution of the loop body `S` to start in state σ_i and terminate in state σ_{i+1}.

It follows that Hoare incorrectness logic is nothing but debugging formalized as a logic. So Hoare incorrectness logic could also be called the debugging logic, to elevate debugging as the computer aided formal method of choice to prove the presence of bugs [10, page 7].

12 Conclusion

A Hoare style (or transformational) logic is an abstract interpretation of a relational semantics, and together with fixpoint induction and Aczel correspondence between set-theoretic fixpoint and deductive system, this leads to the calculational design of the logic proof system, which is semantically sound and complete, by construction.

In this paper, we have considered the abstractions of figure 1. [6] exploits this model theory-based point of view in greater details for many more logics.



collecting semantics $\{[\![\mathtt{S}]\!]\}$
α_C
relational semantics $[\![\mathtt{S}]\!]$
post
postcondition transformer
α_G
antecedent/ consequent pairs
$\mathsf{post}(\supseteq.\subseteq)$
Hoare logic
$\alpha^{\neg}$
Hoare incorrectness logic
$\mathsf{post}(\subseteq, \supseteq)$
incorrectness logic

Fig. 1. Hoare style logic abstractions

We think that, in computer science, the Tarskian/model theoretic approach, which is semantic in nature, is superior to Gentzen/Prawitz proof-theoretic approaches, which are syntactic in nature. Whereas proof calculi are used in mathematics to define truth from which models are derived, we have, in computer science, the advantage that the models of interest are known a priori. They are the semantics of programming languages and systems. Our approach simply exploits this advantage.

Although for most mathematicians and computer scientists, writing a fully formal proof is too pedantic and long-winded to be in common use, the study of program logics is useful to provide the intuition for informal reasonings on program behaviors. Showing that program logic are nothing but an abstract expression of the program semantics strongly supports that intuition. The only difference between logics and semantics is that logic offers ways of making deduction and inference, as made clear by our calculational design which shows that logic = semantics + abstraction + deduction + inference. We have shown

$$\begin{aligned}\text{Hoare logic} &= \text{section } 2 + (4) + (6) + \text{theorem } 2\\ \text{Incorrectness logic} &= \text{section } 2 + (4) + (7) + \text{theorem } 3\\ \text{Hoare incorrectness logic} &= \text{section } 2 + (4) + (8) + \text{theorem } 4\end{aligned}$$

which provides a simple way to explain and compare logics. All the rest is isomorphisms, as shown by the calculational designs only based on equality. This means that the automation of the calculational design might not need the full power of theorem provers since it is essentially rewriting [11].

Disclosure of Interests. This paper is a written version of the talks I gave in London, UK, at Peter O'Hearn Fest on January 14th, 2024 and at POPL'24 on January 19th 2024, with a few simplifications and improved results.

Acknowledgements. I thank Thomas Wies for careful proofreading.

References

1. Aczel, P.: An introduction to inductive definitions. In: Barwise, J. (ed.) Handbook of Mathematical Logic, chap. 7, pp. 739–782. North–Holland, Amsterdam (1977)
2. Cook, S.A.: Soundness and completeness of an axiom system for program verification. SIAM J. Comput. **7**(1), 70–90 (1978). https://doi.org/10.1137/0207005
3. Cook, S.A.: Corrigendum: Soundness and completeness of an axiom system for program verification. SIAM J. Comput. **10**(3), 612 (1981). https://doi.org/10.1137/0210045
4. Cousot, P.: On fixpoint/iteration/variant induction principles for proving total correctness of programs with denotational semantics. In: LOPSTR. Lecture Notes in Computer Science, vol. 12042, pp. 3–18. Springer (2019). https://doi.org/10.1007/978-3-030-45260-5_1
5. Cousot, P.: Principles of Abstract Interpretation. MIT Press, 1 edn. (2021)
6. Cousot, P.: Calculational design of [in]correctness transformational program logics by abstract interpretation. Proc. ACM Program. Lang. **8**(POPL), 175–208 (2024)
7. Cousot, P.: Full version of "calculational design of [in]correctness transformational program logics by abstract interpretation", proc. ACM program. lang. 8, POPL (2024), 7:110:33, https://doi.org/10.1145/3632849. Zenodo p. 66 pages (Dec 2024). https://doi.org/10.5281/zenodo.10439108

8. Cousot, P., Cousot, R.: Constructive versions of Tarski's fixed point theorems. Pacific J. of Math. **82**(1), 43–57 (1979). https://doi.org/10.2140/pjm.1979.82.43
9. Cousot, P., Cousot, R.: Systematic design of program analysis frameworks. In: POPL. pp. 269–282. ACM Press (1979). https://doi.org/10.1145/567752.567778
10. Dijkstra, E.W.: Notes on structured programming. Tech. Rep. T.H.-Report 70-WSK-03, Department of Mathematics, Technological University Eindhoven, The Netherlands (Apr 1970), https://www.cs.utexas.edu/users/EWD/ewd02xx/EWD249.PDF
11. Durán, F., Eker, S., Escobar, S., Martí-Oliet, N., Meseguer, J., Rubio, R., Talcott, C.L.: Programming and symbolic computation in Maude. J. Log. Algebraic Methods Program. **110** (2020)
12. Floyd, R.W.: Assigning meaning to programs. In: Schwartz, J. (ed.) Proc. Symp. in Applied Math., vol. 19, pp. 19–32. Amer. Math. Soc. (1967). https://doi.org/10.1007/978-94-011-1793-7_4
13. Hoare, C.A.R.: An axiomatic basis for computer programming. Commun. ACM **12**(10), 576–580 (1969). https://doi.org/10.1145/363235.363259
14. Kahn, G.: Natural semantics. In: STACS. Lecture Notes in Computer Science, vol. 247, pp. 22–39. Springer (1987). https://doi.org/10.1007/BFB0039592
15. Karp, C.R.: Languages with expressions of infinite length. North–Holland, Amsterdam (1964)
16. Naur, P.: Proofs of algorithms by general snapshots. BIT **6**, 310–316 (1966). https://doi.org/10.1007/BF01966091
17. O'Hearn, P.W.: Incorrectness logic. Proc. ACM Program. Lang. **4**(POPL), 10:1–10:32 (2020). https://doi.org/10.1145/3371078
18. Park, D.M.R.: Fixpoint induction and proofs of program properties. In: Mitchie, D., Meltzer, B. (eds.) Machine Intelligence Volume 5, chap. 3, pp. 59–78. Edinburgh Univ. Press (1969)
19. Tarski, A.: A lattice theoretical fixpoint theorem and its applications. Pacific J. of Math. **5**, 285–310 (1955). https://doi.org/10.2140/pjm.1955.5.285
20. Turing, A.: Checking a large routine. In: Report of a Conference on High Speed Automatic Calculating Machines. pp. 67–69. University of Cambridge Mathematical Laboratory, Cambridge, England (1949 [1950]), https://turingarchive.kings.cam.ac.uk/publications-lectures-and-talks-amtb/amt-b-8
21. de Vries, E., Koutavas, V.: Reverse Hoare logic. In: SEFM. Lecture Notes in Computer Science, vol. 7041, pp. 155–171. Springer (2011). https://doi.org/10.1007/978-3-642-24690-6_12