HOARE LOGIC: FROM FIRST-ORDER TO PROPOSITIONAL FORMALISM [*]

JERZY TIURYN[†]

1. Introduction

Hoare Logic, intoduced by C.A.R. Hoare (Hoare, 1969), is a precursor of Dynamic Logic. It was one of the first formal verification systems designed for proving partial correctess assertions (PCAs) of deterministic **while** programs. It is related to the invariant assertion method of R. Floyd (Floyd, 1967). Both Hoare Logic and Dynamic Logic are examples of what is called the *exogenous* approach to the modal logic of programs. This means that programs are explicit and are part of well formed expressions, the logic is dealing with. It should be contrasted with another approach, called *endogenous*, which is exemplified by Temporal Logic (Pnueli, 1977). In the latter approach the program is fixed and it is viewed as part of the structure in which the logic is interpreted. The interested reader is referred to in-depth surveys on Hoare Logic (Apt, 1981; Apt and Olderog, 1991) and on Temporal Logic (Emerson, 1990; Gabbay et al., 1994).

It follows from complexity considerations that there is no sound and complete proof system capable of deriving all valid PCAs. In particular Hoare Logic is incomplete — in general the intermediate assertions and loop invariants need not be first-order expressible. As shown by S. Cook (Cook, 1978) Hoare Logic is relatively complete, i.e. complete over sufficiently expressive structures. The expressiveness conditions can be stated as first-order definability of weakest liberal preconditions.

Most investigations in Hoare Logic are carried out in the context of first-order language. However, some properties of the formal system can be obscured by restricting attention to a very special form of atomic programs like an assignment statement. A basic question naturally arises: to which

[*] This work was supported by KBN Grant No. 7 T11C 028 20.

[†] Institute of Informatics, Warsaw University, ul. Banacha 2, 02-097 Warsaw, Poland. Email: tiuryn@mimuw.edu.pl

extend the metamathematical properties of Hoare Logic are a consequence of choosing assignment statements as the atomic programs? For this reason one can consider an investigation of Hoare Logic on the propositional level. Propositional Hoare Logic (PHL) was introduced by D. Kozen (Kozen, 1999). It is subsumed by other propositional program logics such as Propositional Dynamic Logic (PDL) (Fischer and Ladner, 1977), or Kleene Algebra with Tests (KAT) (Kozen, 1997). The Hoare PCA $\{b\}\,p\,\{c\}$ is expressed in PDL by the formula $b \rightarrow [p]\,c$ and in KAT by the equation $bp\bar{c} = 0$. The weakes liberal precondition of p with respect to c is expressed in PDL by $[p]\,c$.

As we will see determinig the deductive strength of the original Hoare rules in a propositional context sheds some light on the boundary between Hoare logic proper and the expressiveness assumptions on the underlying domain. Instead of studying derivability in PHL of a single PCA $\{b\}\,p\,\{c\}$ we are concerned with derivability of rules of the form

$$\frac{\{b_1\}\,p_1\,\{c_1\}\ldots\{b_n\}\,p_n\,\{c_n\}}{\{b\}\,p\,\{c\}}, \tag{1}$$

where $p_1,\ldots,p_n$ are programs and $b_1,\ldots,b_n,c_1,\ldots,c_n,b,c$ are propositions. The case of derivability of a single PCA is obtained when $n = 0$ in (1).

In the present exposition we work with programs being regular expressions, rather than **while** programs. Working with the former class of programs does not lead to sacrificing the computational power and it results in simpler rules of inference.

The aim of this paper is to introduce the student to PHL. The paper is organized as follows. In Section 2 we introduce some notation including the class of deterministic **while** programs and their semantics. We also discuss briefly the notion of the weakest liberal precondition. Section 3 is devoted to recalling some basic facts about the first-order Hoare Logic. In particular we discuss there the issues of incompleteness and relative completeness. Derivability and admissibility of rules is also briefly discussed here. Then in Section 4 we introduce Propositional hoare Logic. We show that every valid rule (1) with atomic programs $p_1,\ldots,p_n$ in the premises is derivable in PHL. Examples show that the atomicity assumption in the above result is essential. We also show that if PHL is suitably extended by introducing propositional version of weakes liberal preconditions, every valid rule (1) becomes derivable. Both these results are due to D. Kozen and the author of the present paper (Kozen and Tiuryn, 2000), We decided to include brief sketches of the proofs of these results since the publication of (Kozen and Tiuryn, 2000) is not easily available. Also the details of the presentation of the second result slightly differ from the original publication. We also mention in Section 4 the issue of complexity of deciding validity of rules in PHL.

2. Preliminaries

Let us fix a signature Σ. **While** programs are build from *tests*, being quantifier-free first-order formulas; and *assignment statements* $x := t$ with help of the following programming constructs:

- *Composition*: $\alpha\,;\,\beta$.
- *Conditional*: **if** φ **then** α **else** β.
- *Iteration*: **while** φ **do** α.

Given a Σ-structure $\mathfrak{A}$. The *meaning* of a program α in $\mathfrak{A}$ is a binary *input-output* relation $\mathfrak{m}_{\mathfrak{A}}(\alpha) \subseteq S^{\mathfrak{A}} \times S^{\mathfrak{A}}$.

- $\mathfrak{m}_{\mathfrak{A}}(x := t) \overset{\text{def}}{=} \{(u, u[x/u(t)]) \mid u \in S^{\mathfrak{A}}\}$.
- $\mathfrak{m}_{\mathfrak{A}}(\alpha\,;\,\beta) \overset{\text{def}}{=} \mathfrak{m}_{\mathfrak{A}}(\alpha) \circ \mathfrak{m}_{\mathfrak{A}}(\beta)$.
- $\mathfrak{m}_{\mathfrak{A}}(\textbf{if } \varphi \textbf{ then } \alpha \textbf{ else } \beta) \overset{\text{def}}{=} \mathfrak{m}_{\mathfrak{A}}(\varphi) \circ \mathfrak{m}_{\mathfrak{A}}(\alpha) \cup \mathfrak{m}_{\mathfrak{A}}(\neg\varphi) \circ \mathfrak{m}_{\mathfrak{A}}(\beta)$.
- $\mathfrak{m}_{\mathfrak{A}}(\textbf{while } \varphi \textbf{ do } \alpha) \overset{\text{def}}{=} (\mathfrak{m}_{\mathfrak{A}}(\varphi) \circ \mathfrak{m}_{\mathfrak{A}}(\alpha))^* \circ \mathfrak{m}_{\mathfrak{A}}(\neg\varphi)$.

2.1. WEAKEST LIBERAL PRECONDITION

The concept of weakest liberal precondition was introduced in 1975 by E.W. Dijkstra (Dijkstra, 1975). It played an important role in the area of program semantics. A *weakest liberal precondition* of a program α w.r.t. a formula ψ in a structure $\mathfrak{A}$ is the following predicate

$$WLP_{\mathfrak{A}}(\alpha,\ \psi) \overset{\text{def}}{=} \{u \in S^{\mathfrak{A}} \mid \forall v\ (u,v) \in \mathfrak{m}_{\mathfrak{A}}(\alpha) \Longrightarrow v \in \mathfrak{m}_{\mathfrak{A}}(\psi)\}.$$

This predicate is usually not first-order definable. A structure $\mathfrak{A}$ for which $WLP_{\mathfrak{A}}(\alpha,\ \psi)$ is first-order definable for all programs α and formulas ψ is called *expressive.*

Weakest liberal precondition for α w.r.t. ψ is expressed in Dynamic Logic by the *necessity statement* $[\alpha]\psi$, i.e. we have for every structure $\mathfrak{A}$,

$$WLP_{\mathfrak{A}}(\alpha,\ \psi) = \mathfrak{m}_{\mathfrak{A}}([\alpha]\psi).$$

The basic properties of weakest liberal preconditions are collected in the following result, whose proof can be safely left for the reader. This result can be viewed as an equivalent definition of the weakest liberal precondition.

THEOREM 2.1. *WLP*$(-,\ -)$ *is the least predicate satisfying the following equivalences.*

1. $\models WLP(x := t,\ \psi) \iff \psi[x/t]$.
2. $\models WLP(\textbf{if}\ \sigma\ \textbf{then}\ \alpha\ \textbf{else}\ \beta,\ \psi) \iff (\sigma\wedge WLP(\alpha,\ \psi))\vee(\neg\sigma\wedge WLP(\beta,\ \psi))$.
3. $\models WLP(\alpha\,;\ \beta,\ \psi) \iff WLP(\alpha,\ WLP(\beta,\ \psi))$.
4. $\models WLP(\textbf{while}\ \sigma\ \textbf{do}\ \alpha,\ \psi) \iff (\neg\sigma\wedge\psi)\vee(\sigma\wedge WLP(\alpha,\ WLP(\textbf{while}\ \sigma\ \textbf{do}\ \alpha,\ \psi))$.

3. First-Order Hoare Logic, HL

A *partial correctness assertion* is a triple $\{\varphi\}\ \alpha\ \{\psi\}$, where α is a program and φ, ψ are first-order formulas. It expresses the property that if the program α is run in a state satisfying the *precondition* φ, then upon termination it will satisfy the *postcondition* ψ. Termination of α is not guaranteed, though. More formally, for a structure $\mathfrak{A}$,

$$\mathfrak{A} \vDash \{\varphi\}\ \alpha\ \{\psi\} \overset{\text{def}}{\iff}$$
$$\text{for all } u, v \in S^{\mathfrak{A}},\ \text{if } \mathfrak{A}, u \vDash \varphi \text{ and } (u,v) \in \mathfrak{m}_{\mathfrak{A}}(\alpha),\ \text{then } \mathfrak{A}, v \vDash \psi.$$

The next result connects partial correctness assertions with weakest liberal preconditions. It follows immediately from the definitions.

PROPOSITION 3.1. *For all formulas φ, ψ and every program α,*

$$\models \{\varphi\}\ \alpha\ \{\psi\} \iff \varphi \to WLP(\alpha,\ \psi).$$

A PCA $\{\varphi\}\,\alpha\,\{\psi\}$ is *valid*, denoted $\models \{\varphi\}\,\alpha\,\{\psi\}$, if it holds in all structures. We cannot hope to have a sound proof system capable of deriving all valid PCAs. The reason is that the set of all valid PCAs is too complex.

THEOREM 3.2. **(Complexity of PCA Validity)**
For sufficiently rich signatures Σ, the set of all valid PCAs is Π^0_2-complete.

Since the set of all valid PCAs in a given structure $\mathfrak{A}$ is at least as complex as the first-order theory of $\mathfrak{A}$ we assume that the latter is is given as an oracle in the system we are going to introduce.

$$\textbf{(Assignment)}\quad \{\varphi[x/e]\}\; x := e\; \{\varphi\}$$

$$\textbf{(Composition)}\quad \frac{\{\varphi\}\;\alpha\;\{\xi\},\quad \{\xi\}\;\beta\;\{\psi\}}{\{\varphi\}\;\alpha\,;\,\beta\;\{\psi\}}$$

$$\textbf{(Conditional)}\quad \frac{\{\varphi\wedge\sigma\}\;\alpha\;\{\psi\},\quad \{\varphi\wedge\neg\sigma\}\;\beta\;\{\psi\}}{\{\varphi\}\;\textbf{if}\;\sigma\;\textbf{then}\;\alpha\;\textbf{else}\;\beta\;\{\psi\}}$$

$$\textbf{(While)}\quad \frac{\{\varphi\wedge\sigma\}\;\alpha\;\{\varphi\}}{\{\varphi\}\;\textbf{while}\;\sigma\;\textbf{do}\;\alpha\;\{\varphi\wedge\neg\sigma\}}$$

$$\textbf{(Weakening)}\quad \frac{\mathfrak{A}\models\varphi'\to\varphi,\quad \{\varphi\}\;\alpha\;\{\psi\},\quad \mathfrak{A}\models\psi\to\psi'}{\{\varphi'\}\;\alpha\;\{\psi'\}}$$

Figure 1. HL, Hoare Logic over a structure $\mathfrak{A}$.

We denote by $\vdash_{\mathfrak{A}} \{\varphi\}\;\alpha\;\{\psi\}$ derivability of the PCA $\{\varphi\}\;\alpha\;\{\psi\}$ in HL over $\mathfrak{A}$.

THEOREM 3.3. **(Soundness)**
For every Σ-structure $\mathfrak{A}$, if $\vdash_{\mathfrak{A}} \{\varphi\}\;\alpha\;\{\psi\}$, then $\mathfrak{A}\models\{\varphi\}\;\alpha\;\{\psi\}$.

It follows from Theorem 3.2 that HL is incomplete. The particular reason for incompleteness of HL is that the *intermediate assertion* ξ in **(Composition)** rule and the *invariant assertion* φ in **(While)** rule need not be first-order definable. To illustrate this let us consider the following example (due to M. Wand (Wand, 1978)). Let $\Sigma = \{f, r\}$, where f is a unary operation symbol and r is a unary relation symbol. Consider the structure $\mathfrak{A} = (A, f^{\mathfrak{A}}, r^{\mathfrak{A}})$, where

$$A = \{a_i \mid i\in\mathbb{N}\}\cup\{b_i \mid i\in\mathbb{N}\},$$
$$r^{\mathfrak{A}} = \{a_{k^2} \mid k\in N\},$$

and $f^{\mathfrak{A}}$ is defined as follows (x stands here for a, or b)

$$f^{\mathfrak{A}}(x_i) = \begin{cases} x_0 & \text{if } i = 0,\\ x_{i-1} & \text{if } i > 0.\end{cases}$$

Hence f behaves in $\mathfrak{A}$ like predecessor (on two copies on natural numbers), and r defines an infinite subset of $\{a_i \mid i\in\mathbb{N}\}$ with an increasing distance between two consequtive elements in this subset.

Clearly we have $\mathfrak{A} \models \{r(x)\}$ **while** $x \neq f(x)$ **do** $x := f(x)\ \{r(x)\}$, but

THEOREM 3.4. (Wand, 1978)
The PCA

$$\{r(x)\}\ \textbf{while}\ x \neq f(x)\ \textbf{do}\ x := f(x)\ \{r(x)\}$$

is not derivable in $\vdash_{\mathfrak{A}}$.

The reason why the PCA of Theorem 3.4 is not derivable is that, as it is easy to show, derivability of it would imply that the set $\{a_i \mid i \in \mathbb{N}\}$ is first-order definable in $\mathfrak{A}$. However, this set is not first-order definable in $\mathfrak{A}$. The interested reader should try to prove both these claims.

Let us recall that expressive structures are those for which the weakest precondition for every **while** program is first-order definable. The important examples of expressive structures are finite structures and the standard model of arithmetic, the latter is due to the enormous encoding power of arithmetic.

The following well known result is due to S. Cook (Cook, 1978).

THEOREM 3.5. **(Relative Completeness)**
Hoare logic is relatively complete, i.e. for every expressive structure $\mathfrak{A}$*, if* $\mathfrak{A} \models \{\varphi\}\ \alpha\ \{\psi\}$*, then* $\vdash_{\mathfrak{A}} \{\varphi\}\ \alpha\ \{\psi\}$.

Proof: We will not give the full proof of this result since it is well documented in the literature (see, eg. (Winskel, 1993)). One way of proving this result is to show that if we view the weakest liberal precondition as a first order formula, then in all structures $\mathfrak{A}$ we have

$$\vdash_{\mathfrak{A}} \{WLP(\alpha,\ \psi)\}\ \alpha\ \{\psi\}, \tag{2}$$

holds for all formulas ψ and programs α. The proof of (2) is by induction on α. To conclude the proof of Theorem 3.5 we observe that if $\mathfrak{A}$ is an expressive structure then (2) is obtainable by a legal derivation (i.e. all pre- and postconditions are first-order formulas) and if $\mathfrak{A} \models \{\varphi\}\ \alpha\ \{\psi\}$ holds, then by Proposition 3.1 and weakening applied to (2) we conclude that

$$\vdash_{\mathfrak{A}} \{\varphi\}\ \alpha\ \{\psi\}.$$

■

Let us now briefly discuss the issue of admissibility vs. derivability of rules. Given a rule of the form

$$(\mathbf{R}) \quad \frac{\{\varphi_1\}\ \alpha_1\ \{\psi_1\} \quad \ldots \quad \{\varphi_n\}\ \alpha_n\ \{\psi_n\}}{\{\varphi\}\ \alpha\ \{\psi\}}.$$

(R) is said to be *admissible* (in Hoare logic) if adding it to the rules of Hoare logic does not increase the set of theorems, i.e. for every structure $\mathfrak{A}$ and for every PCA $\{\varphi\}\,\alpha\,\{\psi\}$, this PCA is derivable in HL extended by **(R)** iff $\vdash_{\mathfrak{A}}$ $\{\varphi\}\alpha\{\psi\}$. A stronger notion is that of derivability. **(R)** is said to *derivable* in Hoare logic if the conclusion $\{\varphi\}\,\alpha\,\{\psi\}$ can be derived in HL (uniformly for all structures $\mathfrak{A}$) from the premises $\{\varphi_1\}\,\alpha_1\,\{\psi_1\}\ldots\{\varphi_n\}\,\alpha_n\,\{\psi_n\}$. Another important notion is that of validity of a rule. **(R)** is said to be *valid* if for every structure $\mathfrak{A}$, if all premises of **(R)** are valid in $\mathfrak{A}$, then the conclusion is valid in $\mathfrak{A}$ as well.

Here is an example of an admissible rule. It will play an important role in the next Section.

PROPOSITION 3.6. *The following rule*

$$(\textbf{And/Or})\quad \frac{\{\varphi_i\}\,\alpha\,\{\psi_j\}\quad i=1,\ldots,m; j=1,\ldots,n}{\{\bigvee_{i=1}^{m}\varphi_i\}\,\alpha\,\{\bigwedge_{j=1}^{n}\psi_j\}}$$

is admissible in Hoare logic.

Proof: The proof is by induction on α. We show that when the premises are derivable in HL, then the conclusion is derivable as well. When α is an assignment statement $x := t$, then we first observe that

$$\vdash_{\mathfrak{A}} \{\varphi\}\,x := t\,\{\psi\} \iff \mathfrak{A} \models \varphi \to \psi[x/t]. \tag{3}$$

We leave the proof of (3) for the reader. It follows from (3) that if $\{\varphi_i\}\alpha\{\psi_j\}$ is derivable, then $\mathfrak{A} \models \varphi_i \to \psi_j[x/t]$, for all $i = 1,\ldots,m$ and $j = 1,\ldots,n$. Thus

$$\mathfrak{A} \models \bigvee_{i=1}^{m}\varphi_i \to \bigwedge_{j=1}^{n}\psi_j[x/t]$$

and again by (3) we obtain the conclusion.

The case α being a conditional is immediate. If α is a composition $\beta\,;\,\gamma$ and if $\xi_{i,j}$ is the intermediate assertion for $\vdash_{\mathfrak{A}} \{\varphi_i\}\,\beta\,;\,\gamma\,\{\psi_j\}$, then $\bigwedge_{i,j}\xi_{i,j}$ is the intermediate assertion for $\vdash_{\mathfrak{A}} \{\bigvee_{i=1}^{m}\varphi_i\}\,\beta\,;\,\gamma\,\{\bigwedge_{j=1}^{n}\psi_j\}$.

Finally let us consider the case of α being the iteration **while** σ **do** β. Assume that $\vdash_{\mathfrak{A}} \{\varphi_i\}\,\alpha\,\{\psi_j\}$. Hence there is an invariant $\xi_{i,j}$ such that

$$\varphi_i \to \xi_{i,j} \tag{4}$$

$$\xi_{i,j} \wedge \neg\sigma \to \psi_j \tag{5}$$

and

$$\vdash_{\mathfrak{A}} \{\xi_{i,j} \wedge \sigma\}\,\beta\,\{\xi_{i,j}\}.$$

Let

$$\xi \overset{\text{def}}{\iff} \bigvee_i \bigwedge_j \xi i, j.$$

By the induction hypothesis we have

$$\vdash_{\mathfrak{A}} \{\sigma \wedge \bigvee_{i,j} \xi_{i,j}\}\, \beta\, \{\bigwedge_{i,j} \xi_{i,j}\}.$$

Since $\bigvee_i \bigwedge_j \xi_{i,j} \rightarrow \bigvee_{i,j} \xi_{i,j}$ and $\bigwedge_{i,j} \xi_{i,j} \rightarrow \bigvee_i \bigwedge_j \xi_{i,j}$, by weakening we obtain

$$\vdash_{\mathfrak{A}} \{\sigma \wedge \xi\}\, \beta\, \{\xi\}.$$

Thus $\vdash_{\mathfrak{A}} \{\xi\}\, \textbf{while}\ \sigma\ \textbf{do}\ \beta\, \{\xi \wedge \neg\sigma\}$. By (4) $\bigvee_i \varphi_i \rightarrow \xi$ and by (5) $\xi \wedge \neg\sigma \rightarrow \bigwedge_j \psi_j$. Hence, by weakening we obtain $\vdash_{\mathfrak{A}} \{\bigvee_{i=1}^{m} \varphi_i\}\, \alpha\, \{\bigwedge_{j=1}^{n} \psi_j\}$. ∎

Consider now the following rule

$$\frac{\{\varphi\}\ \textbf{while}\ \sigma\ \textbf{do}\ \alpha\, \{\psi\}}{\{\varphi \wedge \sigma\}\, \alpha\, \{\neg\sigma \rightarrow \psi\}} \tag{6}$$

How can we argue that the above rule is valid? One way is to show the validity of (6) directly from the semantics of the **while** construct. But we can also show it by refering to the properties of weakest liberal precondition listed in Theorem 2.1. We illustrate the latter method since it will be used in the next section. We will be little informal with our argument. Let $\mathfrak{A}$ be any view the weakest liberal preconditions as first-order formulas. Let $\mathfrak{A}$ be any structure such that

$$\mathfrak{A} \models \{\varphi\}\ \textbf{while}\ \sigma\ \textbf{do}\ \alpha\, \{\psi\}. \tag{7}$$

Claim (2) in the proof of Theorem 3.5 states that $\vdash_{\mathfrak{A}} \{\mathit{WLP}(\alpha,\ \xi)\}\alpha\{\xi\}$ holds for all programs α and all formulas ξ. Let us choose for ξ the formula $\mathit{WLP}(\textbf{while}\ \sigma\ \textbf{do}\ \alpha,\ \psi)$. Thus we have

$$\vdash_{\mathfrak{A}} \{\mathit{WLP}(\alpha,\ \mathit{WLP}(\textbf{while}\ \sigma\ \textbf{do}\ \alpha,\ \psi))\}\alpha\{\mathit{WLP}(\textbf{while}\ \sigma\ \textbf{do}\ \alpha,\ \psi)\}. \tag{8}$$

It follows from Theorem 2.1(4) that

$$\mathfrak{A} \models \mathit{WLP}(\textbf{while}\ \sigma\ \textbf{do}\ \alpha,\ \psi) \rightarrow (\neg\sigma \rightarrow \psi) \tag{9}$$

and

$$\begin{aligned} \mathfrak{A} \models \mathit{WLP}(\textbf{while}\ \sigma\ \textbf{do}\ \alpha,\ \psi) \wedge \sigma \rightarrow & \\ & \mathit{WLP}(\alpha,\ \mathit{WLP}(\textbf{while}\ \sigma\ \textbf{do}\ \alpha,\ \psi)) \end{aligned} \tag{10}$$

It follows from Proposition 3.1 applied to (7) that

$$\mathfrak{A} \models \varphi \rightarrow \mathit{WLP}(\textbf{while}\ \sigma\ \textbf{do}\ \alpha,\ \psi).$$

Thus by (9), (10) and weakening applied to (8) we obtain $\vdash_{\mathfrak{A}} \{\varphi\wedge\sigma\}\alpha\{\neg\sigma \rightarrow \psi\}$. Hence

$$\mathfrak{A} \models \{\varphi \wedge \sigma\}\ \alpha\ \{\neg\sigma \rightarrow \psi\}.$$

4. Propositional Hoare Logic, PHL

In the propositional level of reasoning we start with two sorts of atoms: *atomic programs* and *atomic propositions. Propositions* are constructed from atomic propositions, and **0** (*falsehood*) with help of implication $\rightarrow$. We denote the negation $b \rightarrow \mathbf{0}$ by $\bar{b}$. The truth value **1** is defined as $\bar{\mathbf{0}}$. Disjunction and conjunction are defined in terms of $\rightarrow$ and **0** in the usual way. If C is a finite set of propositions, then $\bigvee C$ denotes the disjunction of its elements. In particular we set $\bigvee \emptyset = \mathbf{0}$. Similarly, $\bigwedge C$ denotes the conjunction of the elements of C and we take $\bigwedge \emptyset = \mathbf{1}$.

As in Propositional Dynamic Logic (see (Harel et al., 2000)), instead of a more traditional conditional and **while** constructs we base our programs on two more fundamental programming constructs: iteration * (a reflexive and transitive closure) and a binary nondeterministic choice + (binary set union). *Programs* is the smallest class of expressions satisfying:

- Every atomic program and every proposition is a program.
- If p, q are programs then the following expressions are programs as well.
  - $p\,;\,q$
  - $p + q$
  - p^*

We add parenthesis when necessary. Because of the propositional level of reasoning we cannot talk about assignment statements — the atomic programs play the role of assignment statements, but we are not restricted to think of an atomic program as an assignment statement. The symbol $\cdot$ is overloaded in our approach: it denotes composition of programs or conjunction of formulas, depending on the context.

We interpret propositions and programs in Kripke frames. A Kripke frame $\mathfrak{K}$ consists of a set of states K and a map $\mathfrak{m}_{\mathfrak{K}}$ associating a subset of K with each atomic proposition and a binary relation on K with each atomic program. The map $\mathfrak{m}_{\mathfrak{K}}$ is extended inductively to compound programs and propositions according to standard rules (see (Harel et al., 2000)). For the sake of completeness of exposition we present it below. For propositions we have the following equations.

- $\mathfrak{m}_{\mathfrak{K}}(\mathbf{0}) = \emptyset$.
- $\mathfrak{m}_{\mathfrak{K}}(b \rightarrow c) = \{s \in K \mid s \notin \mathfrak{m}_{\mathfrak{K}}(b) \text{ or } s \in \mathfrak{m}_{\mathfrak{K}}(c)\}$.

We overload the operator $\mathfrak{m}_{\mathfrak{K}}$ in the sense that the binary relation assigned to a proposition viewed as a program is a partial identity, rather than a set of states, as it is the case when the proposition is viewed in the usual way (i.e. as a proposition). It will be always clear from the context in which sense a given proposition is used in an expression.

- $\mathfrak{m}_{\mathfrak{K}}(b) = \{(s,s) \in K \times K \mid s \in \mathfrak{m}_{\mathfrak{K}}(b)\}$.
- $\mathfrak{m}_{\mathfrak{K}}(p\,;\,q) = \mathfrak{m}_{\mathfrak{K}}(p) \circ \mathfrak{m}_{\mathfrak{K}}(q)$.
- $\mathfrak{m}_{\mathfrak{K}}(p+q) = \mathfrak{m}_{\mathfrak{K}}(p) \cup \mathfrak{m}_{\mathfrak{K}}(q)$.
- $\mathfrak{m}_{\mathfrak{K}}(p^*) = \bigcup_{n\geq 0}(\mathfrak{m}_{\mathfrak{K}}(p))^n$.

We write $\mathfrak{K}, s \vDash b$ for $s \in \mathfrak{m}_{\mathfrak{K}}(b)$ and $s \xrightarrow[\mathfrak{K}]{p} t$ for $(s,t) \in \mathfrak{m}_{\mathfrak{K}}(p)$.

It follows from the above semantics of programs that conditional and **while** constructs are definable in our formalism:

$$\textbf{if } b \textbf{ then } p \textbf{ else } q \overset{\text{def}}{=} bp + \bar{b}q \tag{11}$$

and

$$\textbf{while } b \textbf{ do } p \overset{\text{def}}{=} (bp)^*\bar{b}. \tag{12}$$

The PCA $\{b\}\,p\,\{c\}$ says intuitively that if b holds before executing p, then c must hold after. Formally, the meaning in PHL is the same as the meaning of $b \to [p]\,c$ in PDL: in a state s of a Kripke frame $\mathfrak{K}$,

$$\mathfrak{K}, s \vDash \{b\}\,p\,\{c\} \overset{\text{def}}{\Longleftrightarrow} (\mathfrak{K}, s \vDash b \implies \forall t\ (s \xrightarrow[\mathfrak{K}]{p} t \implies \mathfrak{K}, t \vDash c)).$$

For φ a PCA and Φ a set of PCAs, we write

$$\begin{aligned} \mathfrak{K} \vDash \varphi &\overset{\text{def}}{\Longleftrightarrow} \forall s \in \mathfrak{K} \quad \mathfrak{K}, s \vDash \varphi \\ \mathfrak{K} \vDash \Phi &\overset{\text{def}}{\Longleftrightarrow} \forall \varphi \in \Phi \quad \mathfrak{K} \vDash \varphi \\ \Phi \vDash \varphi &\overset{\text{def}}{\Longleftrightarrow} \forall \mathfrak{K} \quad \mathfrak{K} \vDash \Phi \Longrightarrow \mathfrak{K} \vDash \varphi. \end{aligned}$$

Consider the general form of a rule of inference.

$$\textbf{(R)} \qquad \frac{\{b_1\}\,p_1\,\{c_1\}, \ldots, \{b_n\}\,p_n\,\{c_n\}}{\{b\}\,p\,\{c\}} \tag{13}$$

The rule **(R)** is said to be *valid* if $\{\{b_1\}p_1\{c_1\}, \ldots, \{b_n\}p_n\{c_n\}\} \models \{b\}p\{c\}$. Call a PCA $\{b\}\,p\,\{c\}$ *atomic* if p is an atomic program. The rule **(R)** is said to be *atomic* if all its premises are atomic.

We can rewrite the traditional Hoare rules in the propositional level — they would look exactly the same as in Figure 1, except that the assignment rule would be missing. The pre- and postconditons, as well as tests in the propositional level are propositions. Then the conditional and while rules are replaced by simpler rules, as can be seen in Figure 2. We also need the and/or rule in PHL for reasons which will become clear a bit later.

$$\textbf{(Test)} \quad \{b\}\,c\,\{bc\}$$

$$\textbf{(Composition)} \quad \frac{\{b\}\,p\,\{c\}, \quad \{c\}\,q\,\{d\}}{\{b\}\,p\,;\,q\,\{d\}}$$

$$\textbf{(Choice)} \quad \frac{\{b\}\,p\,\{c\}, \quad \{b\}\,q\,\{c\}}{\{b\}\,p+q\,\{c\}}$$

$$\textbf{(Iteration)} \quad \frac{\{b\}\,p\,\{b\}}{\{b\}\,p^*\,\{b\}}$$

$$\textbf{(Weakening)} \quad \frac{b' \to b, \quad \{b\}\,p\,\{c\}, \quad c \to c'}{\{b'\}\,p\,\{c'\}}$$

$$\textbf{(And/Or)} \quad \frac{\{b\}\,p\,\{c\} \quad b \in B,\ c \in C}{\{\bigvee B\}\,p\,\{\bigwedge C\}}$$

Figure 2. PHL, Propositional Hoare Logic. p, q are programs, b, c are propositions, B, C are finite sets of propositions.

It is immediate to show that all the rules of PHL are valid. Let us observe that without **(And/Or)** rule no atomic PCA is derivable. However, with the help of this rule we can derive a few atomic PCAs: taking $B = \{b\}$ and $C = \emptyset$ we get $\vdash \{b\}\,p\,\{\mathbf{1}\}$; on the other hand, taking $B = \emptyset$ and $C = \{c\}$ we get $\vdash \{\mathbf{0}\}\,p\,\{c\}$. Thus **(And/Or)** rule is not admissible in PHL.

Even with the **(And/Or)** rule very few PCAs are derivable. The reason is that no specific axioms are assumed for atomic PCAs. Thus we investigate derivability of valid atomic rules. Let us recall that derivability in PHL of the rule **(R)** means that the conclusion $\{b\}\,p\,\{c\}$ is derivable in PHL from the additional PCAs $\{b_1\}\,p_1\,\{c_1\}, \ldots, \{b_n\}\,p_n\,\{c_n\}$ treated as extra axioms. For example, translation of the conditional rule under the definition (11) becomes

$$\textbf{(Cond)} \qquad \frac{\{bc\}\,p\,\{d\} \quad \{\bar{b}c\}\,q\,\{d\}}{\{c\}\,bp+\bar{b}q\,\{d\}}.$$

It is derivable in PHL. Indeed, assuming $\{bc\}\, p\, \{d\}$, by **(Test)** and **(Composition)** we obtain $\{c\}\, bp\, \{d\}$. In a similar way we obtain $\{c\}\, \bar{b}p\, \{d\}$. Thus, by (Choice) we get the conclusion $\{c\}\, bp+\bar{b}q\, \{d\}$. In a similar way one shows derivability of the translation of **(While)** rule:

$$\textbf{(Wh)} \qquad \frac{\{bc\}\; p\; \{c\}}{\{c\}\; (bp)^*\bar{b}\; \{\bar{b}c\}}.$$

Atomic rules are potentially interesting for the reason that they express partial correctness of a compound program, subject to partiall correctness assumprions about its atomic components. The rules of PHL can be viewed in this way. For example, the **(Composition)** rule says that the composition of two programs is partially correct under the assumption of suitable partial correctness assertions for both the programs. Hence one of the issues of completeness of PHL can be expressed as follows. Given an atomic rule which is valid. Is it derivable in PHL? An affirmative answer is given in the following result.

THEOREM 4.1. (Kozen and Tiuryn, 2000)
Every valid atomic rule of the form (13) is derivable in PHL.

Proof: We sketch the main steps of the proof of Theorem 4.1. For a finite set Φ of PCAs an a PCA φ we write $\Phi \vdash \varphi$ if the conclusion φ is derivable from the premises Φ in PHL. Suppose Φ is a set of atomic PCAs and φ a PCA such that $\Phi \nvdash \varphi$. A Kripke frame $\mathfrak{K}$ is constructed such that $\mathfrak{K} \vDash \Phi$ but $\mathfrak{K} \nvDash \varphi$.

Let us call an *atom* any maximal propositionally consistent conjuction of atomic propositions, which occur in Φ or φ, or their negations. Atoms are denoted $\alpha, \beta, \gamma, \ldots$. We identify an atom α with the conjunction of all formulas in α. The set K of states of our Kripke frame is the set of all atoms. For propositons b, c we write $b \leq c$ if $b \to c$ is a propositional tautology.

For atomic programs a and atomic propositions b, define

$$\begin{aligned} \mathfrak{m}_{\mathfrak{K}}(a) &\stackrel{\text{def}}{=} \{(\alpha,\beta) \mid \Phi \nvdash \{\alpha\}\; a\; \{\bar{\beta}\}\} \\ \mathfrak{m}_{\mathfrak{K}}(b) &\stackrel{\text{def}}{=} \{\alpha \mid \alpha \leq b\}. \end{aligned}$$

Since all premises in Φ are atomic, it follows that $\mathfrak{K} \models \Phi$.

To conclude the proof we show that for every program p and for all atoms α, β,

$$\Phi \nvdash \{\alpha\}\; p\; \{\bar{\beta}\} \implies \alpha \xrightarrow[\mathfrak{K}]{p} \beta. \tag{14}$$

One shows the contrapositive of (14) by induction on p. Having proved (14) let us assume that $\Phi \nvdash \{b\}\; p\; \{c\}$. By the **(And/Or)** rule, there must exist

$\alpha \leq b$ and $\beta \leq \bar{c}$ such that $\Phi \not\models \{\alpha\}\, p\, \{\bar{\beta}\}$. Hence, by (14) we conclude that $\alpha \xrightarrow[\mathfrak{K}]{p} \beta$, i.e. $\mathfrak{K} \not\models \{b\}\, p\, \{c\}$. ∎

The reader may have started wondering whether thet reason for completeness in Theorem 4.1 is that the set of all valid atomic rules is relatively small. i.e. simple. It follows from the next result that in fact it is not the case: decidading validity of an atomic rule is as difficult as deciding validity of a quantified propositional formula.

THEOREM 4.2. (Kozen, 1999)
The set of all valid atomic rules of the form (13) is PSPACE-complete.

The original proof of the lower bound in the above theorem was by direct encoding of polynomial space-bounded deterministic Turing machines. A shorter proof can be found in (Cohen and Kozen, 2000), where the reduction is from the universality problem for nondeterministic finite automata.

Let us observe that the assumption of atomicity of a rule in Theorem 4.1 is essential. Indeed, the following rule is valid

$$\textbf{(Iteration-Reverse)} \qquad \frac{\{b\}\, p^*\, \{c\}}{\{b\}\, p\, \{c\}} \tag{15}$$

but it is not derivable in PHL. The reason is that in the rules of PHL one can never get a conclusion PCA with a program which is simpler than any of the programs occuring in the premises. Hence, in order to obtain completeness for more general rules, we have to enrich the system. We will use the idea of weakest liberal preconditions.

4.1. WEAKEST PRECONDITIONS

We extend our assertion language with formulas of the form

$$b \;\to\; [p_1]\,[p_2]\cdots[p_n]\,c,$$

where b, c are propositions and $p_1, \ldots, p_n$ are regular programs. We call such formulas *extended PCAs*. When $n = 0$, the above expression reduces to the ordinary proposition $b \to c$. In this sense extende PCAs contain propositions. We will abbreviate $\mathbf{1} \to [p_1]\,[p_2]\cdots[p_n]\,c$ by $[p_1]\,[p_2]\cdots[p_n]\,c$.

We assume that in a Kripke frame with each extended PCA a set of states is assigned, i.e. that extended PCAs are assigned truth values in each state of the model. We assume that the interpretation is such that the following properties are satisfied.

$$\begin{aligned}
[s \models [a]\psi] &\leftrightarrow \forall t[(s,t) \in \mathfrak{m}_{\mathfrak{K}}(a) \to t \models \psi] && (16)\\
[p+q]\psi &\leftrightarrow [p]\psi \wedge [q]\psi && (17)\\
[p\,;\,q]\psi &\leftrightarrow [p]\,[q]\psi && (18)\\
[p^*]\psi &\leftrightarrow \psi \wedge [p]\,[p^*]\psi && (19)\\
[b]\psi &\leftrightarrow (b \to \psi). && (20)
\end{aligned}$$

In (16) a is an atomic program. This property establishes the expressiveness of the weakest precondition in the language of extended PCAs. Properties (17-20) are axioms of PDL (see (Harel et al., 2000)) and are related to the basic properties of the weakest liberal preconditions for the first-order case (cf. Theorem 2.1). We will use letters φ, ψ, γ to range over extended PCAs.

Now we enrich PHL in order to get the completeness for arbitrary valid rules.

(Atom) $\{[a]\psi\}\; a\; \{\psi\}$

(Extended PCA Intro) $\dfrac{\{b\}\; p\; \{c\}}{b \to [p]\,c}$

Figure 3. EPHL, Extended Propositional Hoare Logic: PHL, as defined in Fig. 2, augmented with the above two rules. a in **(Atom)** is an atomic program.

PROPOSITION 4.3. *For every Kripke frame $\mathfrak{K}$ satisfying (16-20), the axiom* **(Atom)** *holds in $\mathfrak{K}$ and the rule* **(Extended PCA Intro)** *is valid in $\mathfrak{K}$.*

Proof: Validity of **(Atom)** follows immediately from implication $\to$ in (16). Validity of **(Extended PCA Intro)** is proved by induction on p. For an atomic p it follows from the implication $\leftarrow$ in (16). Each of the induction steps is handled by one of the properties (17-20). The proof is routine and we leave the details for the interested reader. ∎

The next resuls says that the axiom **(Atom)** can be extended to all programs. Let us denote by $\vdash_{EPHL} \{\varphi\}\; p\; \{\psi\}$ derivability in EPHL the PCA $\{\varphi\}\, p\, \{\psi\}$.

PROPOSITION 4.4. *For every program p and for every extended PCA ψ, the PCA $\{[p]\psi\}\, p\, \{\psi\}$ is derivable in EPHL, i.e. $\vdash_{EPHL} \{[p]\psi\}\, p\, \{\psi\}$.*

Proof: The proof is by induction on p. The base step is just **(Atom)**. We just show the induction step for p being a composition $q_1\,;\,q_2$. The other cases, being similar, are left for the reader. By the induction hypothesis we have

$$\vdash_{EPHL} \{[q_1]\,[q_2]\psi\}\; q_1\; \{[q_2]\psi\}$$

and

$$\vdash_{EPHL} \{[q_2]\psi\}\; q_2\; \{\psi\}.$$

By **(Composition)** we obtain

$$\vdash_{EPHL} \{[q_1]\,[q_2]\psi\}\; q_1\,;\,q_2\; \{\psi\}.$$

By (18) and weakening we obtain

$$\vdash_{EPHL} \{[q_1\,;\,q_2]\psi\}\; q_1\,;\,q_2\; \{\psi\}.$$

∎

Let us show how to derive in EPHL the rule **(Iteration-Reverse)** (see (15)). Assume $\{b\}\, p^*\, \{c\}$, by **(Extended PCA Intro)** we get

$$b \to [p^*]\,c. \tag{21}$$

Since by (19)

$$[p^*]\,c \;\leftrightarrow\; c \land [p]\,[p^*]\,c, \tag{22}$$

it follows by propositional reasoning that

$$[p^*]\,c \to c \tag{23}$$

and by (21) and (22)

$$b \to [p]\,[p^*]\,c. \tag{24}$$

Now, we have an instance of **(Atom)**:

$$\{[p]\,[p^*]\,c\}\; p\; \{[p^*]\,c\}.$$

Thus by (23), (24) and weakening we obtain

$$\{b\}\; p\; \{c\}.$$

THEOREM 4.5. (Kozen and Tiuryn, 2000)
Every valid rule of the form (13) is derivable in EPHL.

Proof: For a given set X of extended PCAs we define the *Fisher–Ladner closure* $FL(X)$ in a similar way as in PDL (see (Harel et al., 2000)). A set X of extended PCAs is said to be *closed* if it satisfies the following closure properties:

- $b \to \psi \in X \quad \Longrightarrow \quad b \in X$ and $\psi \in X$
- $\mathbf{0} \in X$
- $[p+q]\psi \in X \quad \Longrightarrow \quad [p]\psi \in X$ and $[q]\psi \in X$
- $[p\,;\,q]\psi \in X \quad \Longrightarrow \quad [p][q]\psi \in X$ and $[q]\psi \in X$
- $[p^*]\psi \in X \quad \Longrightarrow \quad \psi \in X$ and $[p][p^*]\psi \in X$
- $[b]\psi \in X \quad \Longrightarrow \quad b \to \psi \in X$
- $[a]\psi \in X \quad \Longrightarrow \quad \psi \in X.$

$FL(X)$ is the least closed set containing X. The important property of this closure is that for a finite set X of extended PCAs, $FL(X)$ is again a finite set of extended PCAs.

Let $\Phi = \{b_1 \to [p_1]c_1, \ldots b_n \to [p_n]c_n\}$, where $\{b_1\}p_1\{c_1\}, \ldots, \{b_n\}p_n\{c_n\}$ are the premises of the rule (13).

An *atom* α is a set of formulas of $FL(\Phi)$ and their negations satisfying the following properties:

(i) for each $\psi \in FL(\Phi)$, exactly one of ψ, $\bar{\psi} \in \alpha$
(ii) for $b \to \psi \in FL(\Phi)$, $b \to \psi \in \alpha \iff (b \in \alpha \implies \psi \in \alpha)$
(iii) $\mathbf{0} \notin \alpha$
(iv) for $[p+q]\psi \in FL(\Phi)$, $[p+q]\psi \in \alpha \iff [p]\psi \in \alpha$ and $[q]\psi \in \alpha$
(v) for $[p\,;\,q]\psi \in FL(\Phi)$, $[p\,;\,q]\psi \in \alpha \iff [p][q]\psi \in \alpha$
(vi) for $[p^*]\psi \in FL(\Phi)$, $[p^*]\psi \in \alpha \iff \psi \in \alpha$ and $[p][p^*]\psi \in \alpha$
(vii) for $[b]\psi \in FL(\Phi)$, $[b]\psi \in \alpha \iff b \to \psi \in \alpha$
(viii) if $\{b\}\,p\,\{c\} \in \Phi$, then $b \to [p]c \in \alpha$.

Thus atoms represent consistency conditions not only implied by propositional logic, but also the properties (17-20). It follows that since $FL(\Phi)$ is finite the set of all atoms is finite too. Let K be the set of all atoms.

Now we can construct a finite Kripke frame $\mathfrak{K}$ with states K. We define

$$\begin{aligned}
\mathfrak{m}_{\mathfrak{K}}(a) &\stackrel{\text{def}}{=} \{(\alpha,\beta) \mid \forall [a]\psi \in FL(\Phi)\ ([a]\psi \in \alpha \Longrightarrow \psi \in \beta)\} \\
\mathfrak{m}_{\mathfrak{K}}(b) &\stackrel{\text{def}}{=} \{\alpha \mid b \in \alpha\} \\
\mathfrak{m}_{\mathfrak{K}}([p]\psi) &\stackrel{\text{def}}{=} \{\alpha \mid [p]\psi \in \alpha\}
\end{aligned}$$

for atomic programs a, atomic propositions b, and extended PCAs of the form $[p]\psi$. The meaning function $\mathfrak{m}_{\mathfrak{K}}$ is lifted to compound programs and propositions according to the usual inductive rules. It is immediate to show that for every proposition b and a state α,

$$b \in \alpha \iff \alpha \models b.$$

In the above definition of the frame $\mathfrak{K}$, formulas of the form $[p]\,\psi$ occuring in $FL(\Phi)$ are treated as atomic. However, if for a given extended PCA $\psi = b \;\rightarrow\; [p_1]\,[p_2] \cdots [p_n]\,c$ we denote by $\widehat{\psi}$ the PCA $\{b\}\,p_1;\ldots;p_n\,\{c\}$, then the following property can be proved in $\mathfrak{K}$: for every $\psi \in FL(\Phi)$ and for every state α,

$$\psi \in \alpha \Longrightarrow \alpha \models \widehat{\psi}. \tag{25}$$

To prove (25) we first show by induction on p that for an extended PCA $[p]\,\psi \in FL(\Phi)$ and atoms α, β,

$$[p]\,\psi \in \alpha \text{ and } \alpha \xrightarrow[\mathfrak{K}]{p} \beta \;\Longrightarrow\; \psi \in \beta.$$

Then (25) follows by a simple induction on ψ.

Let $\Psi = \{\{b_1\}\,p_1\,\{c_1\},\ldots,\{b_n\}\,p_n\,\{c_n\}\}$ be the set of premises of the rule (13). It follows from (25) that $\mathfrak{K} \models \Psi$.

Let us recall that, as in the proof of Theorem 4.1, we identify an atom α with the conjuction of all formulas in α. To complete the proof of Theorem 4.5 we prove that for every program p and all atoms α, β,

$$\Psi \not\vdash_{EPHL} \{\alpha\}\,p\,\{\bar{\beta}\} \;\Longrightarrow\; \alpha \xrightarrow[\mathfrak{K}]{p} \beta. \tag{26}$$

The proof of the statement contrapositive to (26) is by induction on p. For the base case we use the axiom **(Atom)**, **(Weakening)** and the meaning function for atomic programs.

Now, if $\Psi \not\vdash_{EPHL} \{b\}p\{c\}$, then it is easy to show by the **(And/Or)** rule that there exist states α, β such that $\alpha \models b$, $\beta \models \bar{c}$ and $\Psi \not\vdash_{EPHL} \{\alpha\}\,p\,\{\bar{\beta}\}$. Thus by (26) we obtain $\alpha \xrightarrow[\mathfrak{K}]{p} \beta$ and therefore $\alpha \not\models \{b\}p\{c\}$. This completes the proof. ∎

References

Apt, K. R.: 1981, 'Ten years of Hoare's logic: a survey—part I'. *ACM Trans. Programming Languages and Systems* **3**, 431–483.

Apt, K. R. and E.-R. Olderog: 1991, *Verification of Sequential and Concurrent Programs.* Springer-Verlag.

Cohen, E. and D. Kozen: 2000, 'A Note on the complexity of propositional Hoare logic'. *Trans. Computational Logic* **1**(1), 171–174.

Cook, S. A.: 1978, 'Soundness and completeness of an axiom system for program verification'. *SIAM J. Comput.* **7**, 70–80.

Dijkstra, E.: 1975, 'Guarded Commands, Nondeterminacy and Formal Derivation of Programs'. *CACM* **18**(8), 453–457.

Emerson, E. A.: 1990, 'Temporal and modal logic'. In: J. van Leeuwen (ed.): *Handbook of theoretical computer science*, Vol. B: formal models and semantics. Elsevier, pp. 995–1072.

Fischer, M. J. and R. E. Ladner: 1977, 'Propositional modal logic of programs'. In: *Proc. 9th Symp. Theory of Comput.* pp. 286–294.

Floyd, R.: 1967, 'Assigning Meaning to Programs'. In: J. Schwartz (ed.): *Proc. Symp. in Applied Mathematics.* pp. 19–32.

Gabbay, D., I. Hodkinson, and M. Reynolds: 1994, *Temporal Logic: Mathematical Foundations and Computational Aspects.* Oxford University Press.

Harel, D., D. Kozen, and J. Tiuryn: 2000, *Dynamic Logic.* Cambridge, MA: MIT Press.

Hoare, C.: 1969, 'An Axiomatic Basis for Computer Programming'. *CACM* **12**(10), 576–580.

Kozen, D.: 1997, 'Kleene algebra with tests'. *Transactions on Programming Languages and Systems* **19**(3), 427–443.

Kozen, D.: 1999, 'On Hoare logic and Kleene algebra with tests'. In: *Proc. Conf. Logic in Computer Science (LICS'99).* pp. 167–172.

Kozen, D. and J. Tiuryn: 2000, 'On the completeness of propositional Hoare logic'. In: J. Desharnais (ed.): *Proc. 5th Int. Seminar Relational Methods in Computer Science (RelMiCS 2000).* pp. 195–202.

Pnueli, A.: 1977, 'The temporal logic of programs'. In: *Proc. 18th Symp. Found. Comput. Sci.* pp. 46–57.

Wand, M.: 1978, 'A new incompleteness result for Hoare's system'. *J. Assoc. Comput. Mach.* **25**, 168–175.

Winskel, G.: 1993, *The formal semantics of programming languages.* MIT Press.