Is “Some-Other-Time” Sometimes Better Than “Sometime” for Proving Partial Correctness of Programs? 1

Abstract. The main result of this paper belongs to the field of the comparative study of program verification methods as well as to the field called nonstandard logics of programs. We compare the program verifying powers of various well-known temporal logics of programs, one of which is the Intermittent Assertions Method, denoted as Bur. Bur is based on one of the simplest modal logics called S5 or “sometime”-logic. We will see that the minor change in this “background” modal logic increases the program verifying power of Bur. The change can be described either technically as replacing the “reflexive version” of S5 with an “irreflexive version”, or intuitively as using the modality “some-other-time” instead of “sometime”. Some insights into the nature of computational induction and its variants are also obtained.

§ 1. Introduction

The main result of the present paper belongs to the field of the comparative study of program verification methods as well as to the field called nonstandard logics of programs (NLP from now on).

One of the things NLP is good for is comparing the program verifying powers of different well-known program verification methods such as Floyd-Hoare method, Burstall’s Intermittent Assertions Method, Pnueli’s Temporal Logics of Programs (e.g. the Nexttime Systems of [17]), Manna and Cooper’s method (for proving total correctness assertions) etc. In [25], which concentrates on proving partial correctness assertions, we proved that Burstall’s Intermittent Assertions Method is strictly stronger than Floyd-Hoare method, and that Burstall’s method is strictly weaker than Pnueli’s Nexttime System with respect to partial correctness assertions. That is, every partial correctness assertion provable by Floyd-Hoare method can be proved by Burstall’s method but not vice versa; and every partial correctness assertion provable by Burstall’s method is provable in Pnueli’s Nexttime System but not vice versa. In [25, 26], Floyd-Hoare’s method was denoted by \( \vdash^F \), Burstall’s method by \( \vdash^{Bur} \), and Pnueli’s Nexttime System by \( \vdash^{Pnu} \), and the Theorem quoted above reads as

\[
\vdash^F < \square \vdash^{Bur} < \square \vdash^{Pnu},
\]

where the symbol \( < \square \) means strictly greater power for proving partial correctness assertions (see § 3.1 of [25]). The main result (Theorem 1) of the

---

1 This project was supported by the Hungarian National Foundation for Scientific Research, Grant No. 1810.
present paper implies that there is a rather naturally defined new program verification method \( \vdash_{Bur}^+ \) the program verifying power of which is strictly between those of \( \vdash_{Bur} \) and \( \vdash_{Pnu} \) (with respect to partial correctness assertions). That is:

\[
\vdash_{Bur} < \square \vdash_{Bur}^+ < \square \vdash_{Pnu}.
\]

Theorem 1 says that a slight and very natural modification of \( \vdash_{Bur} \) produces a new program verification method \( \vdash_{Bur}^+ \) the program verifying power of which is strictly stronger than that of \( \vdash_{Bur} \) for proving partial correctness assertions:

**Theorem 1.** \( \vdash_{Bur} < \square \vdash_{Bur}^+ \).

Recall that Burstall's method \( \vdash_{Bur} \) is based on a multimodal logic the crucial modality of which is called "sometime" (see [6]). The semantical interpretation of sometime is an equivalence relation \( E_2 \) (on the possible states). In particular \( E_2 \) is reflexive. This, of course, means that on the syntactical level (of the formal system \( \vdash_{Bar} \) or of modal logic) an axiom expressing the reflexivity of \( E_2 \) should appear. And indeed, the logical axiom scheme \( \varphi \rightarrow \text{sometime } \varphi \) of \( \vdash_{Bar} \) (axiom (C3) in [17]), called reflexivity axiom in the literature of modal (and temporal) logic, does just that. Now the modified version of \( \vdash_{Bar} \) is obtained, intuitively, from \( \vdash_{Bur} \) by replacing the reflexive notion of an equivalence by the irreflexive one. Formally, \( \vdash_{Bur}^+ \) is obtained from \( \vdash_{Bar} \) by omitting the (above quoted) axiom expressing reflexivity of \( E_2 \) and replacing it with a new logical axiom (Arx in § 2) expressing something in the direction of irreflexivity of \( E_2 \) (the relation interpreting the modality sometime).\(^2\) Thus our Theorem 1 says that the "irreflexive version" \( \vdash_{Bur}^+ \) of \( \vdash_{Bar} \) has strictly greater power than \( \vdash_{Bur} \) for proving partial correctness assertions. The reader interested only in Theorem 1 may skip the rest of § 1.

An easy corollary of Theorem 1 (Corollary 2) solves the problem raised in Figure 2 on p. 333 of [18] and in Figure 1 on p. 16 of [3]. There the question was: When reasoning about programs, is it useful to postulate the successor axioms on the time scale, or are these axioms so trivial (or built-in into the notion of a trace) that they do not help in proving statements about programs? In other words, is there a partial correctness assertion which is not provable from the (so-called computational) induction axioms \( Ind \) while it is provable from \( (Ind + Tsuc) \), where \( Tsuc \) denotes the successor axioms on time. The answer may depend, of course, on the kind of computational induction (i.e. time-induction) we assume. (It is well known that some kind of computational induction is always needed for proving the partial correctness of programs.) In the mathematically exact formalism of NLP (see § 2.2 of [25] or § 4 of [18]), the problem reads as follows:

Is \( Ind < \square (Ind + Tsuc) \) true or not?

\(^2\) In Modal Logic the (reflexive) "sometime-logic" is called \( S5 \) and its irreflexive versions have already been considered, see e.g. [20] p. 55 lines 9–12.