Semantics, Orderings and Recursion in the Weakest Precondition Calculus

Marcello Bonsangue*
CWI,
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands.
Email: marcello@cwi.nl.

Joost N. Kok
Utrecht University, Department of Computer Science,
P.O. Box 80.089, 3508 TB Utrecht, The Netherlands.
Email: joost@cs.ruu.nl.

ABSTRACT An extension of Dijkstra's guarded command language is studied, including sequential composition, demonic choice and a backtrack operator. To guide the intuition about this language we give an operational semantic that relates the initial states with possible outcome of the computations. Next we consider three orderings on this language: a refinement ordering defined by Back, a new deadlock ordering, and an approximation ordering of Nelson. The deadlock ordering is in between the two other orderings. All operators are monotonic in Nelson's ordering, but backtracking is not monotonic in Back's ordering and sequential composition is not monotonic for the deadlock ordering. At first sight recursion can only be added using Nelson's ordering. By extending the fixed point theory we show that, under certain circumstances, least fixed points for non monotonic functions can be obtained by iteration from the least element. This permits us the addition of recursion even using Back's ordering or the deadlock ordering. Furthermore, we give a semantic characterization of the three orderings above by extending the well known duality theory between predicate transformers and Smyth's powerdomain.

Keywords weakest preconditions, predicate transformers, refinement, deadlock, backtracking, recursion, fixed points, fixed point transformations, Smyth powerdomain, Egli-Milner powerdomain.

CONTENTS

1. Introduction
2. Language and Semantics
3. Orderings

*The research of this author was supported by a grant of the Universita' degli Studi di Milano, Italy.
4. Order Theory
4.1 Fixed Points
4.2 Predicate Transformers and Discrete Powerdomains
5. Recursion

1 Introduction

The weakest precondition calculus of Dijkstra identifies statements in the guarded command language with weakest precondition predicate transformers (see [Dij76]). The language was extended to use it as a vehicle for program refinement. Specification constructs were added and a refinement ordering was defined. This approach was introduced in [Bac78, Bac80] and is suited for refinement (see [BvW90, Bac90] and also [MRG88, Mor87]). The refinement ordering can be used to add recursion to the language, but not in a fully compositional way. For example, for each set of guards there is a different conditional command.

Recursion was added in a fully compositional way by Nelson in [Nel87]: the guarded command language was embedded in a language with sequential composition, demonic choice and a backtrack operator in which the operators can be used freely. An ordering is given for which the operators are all monotonic. This ordering is an approximation ordering of the kind used in denotational semantics and does not seem to be suited for refinement. It is defined with the additional notion of weakest liberal preconditions.

Our starting point is the language of [Nel87]. In this language we also have a form of infinite behaviour (a loop construct) and atomic actions that can deadlock (to initiate backtracking). Then we consider three orderings; besides the orderings of Back and Nelson we define a new ordering in between. It is called deadlock ordering because it preserves deadlocks as can be seen from the semantic characterization of the deadlock ordering. In terms of refinement: a normal (non-miraculous) terminating statement is not refined by a miracle in the deadlock ordering.

Only Nelson's ordering is monotonic with respect to all three operators, while the backtrack operator is not monotonic with respect to Back's ordering and the sequential composition is not monotonic for the deadlock ordering. At first sight only Nelson's ordering seems to be suited to add recursion to the full language. But the fact that for Nelson's ordering all the operators are monotonic implies that also recursion can be added with the other two orderings.

In order to show this we extend the fixed point theory. It is well known that a monotone and continuous function from a complete partial order to itself has least fixed point that can be obtained by iteration from the least element. This result was extended at first by Hitchcock and Park [HP72] showing that for a function from a complete partial order to itself is enough to be monotone in order to have a least fixed point. Then Apt and Plotkin [AP86] have shown that the least fixed point property can be transferred, via a commutative diagram, to monotone functions from a partial order to itself. Finally, in [BK92] we show that the least fixed point property can be transferred, via a commutative