ELSEVIER

Available online at www.sciencedirect.com

SCIENCE DIRECT

Electronic Notes in Theoretical Computer Science

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Abstract

Abstract

The formal semantics of a programming language consists of assigning to every program of the language an element of a mathematical structure. In this monograph we study the relationship between two different approaches to define the semantics of a program, namely the denotational and the axiomatic one.

The denotational semantics characterizes programs as elements of some mathematical domain in a compositional way: the semantics of a language construct is defined in terms of its components. Due to the possibility of self-application given by some programming languages, the semantic domain must sometimes be defined in a recursive way.

The axiomatic semantics characterizes programs in a logical framework intended for reasoning about programs properties: computations are expressed by relating programs to assertions about their behaviour.

We study different transformations which ensure the correctness of one semantics in terms of the other. These transformations form dualities rather than equivalences. This is due to the fact that denotationally programs are identified with functions which transform states on the input space to (sets of) states of the output space, whereas axiomatically programs can be expressed as functions which transform predicates on the output space to predicates on the input space.

The dualities between the denotational and the axiomatic views of a program are topological because they are set in a topological framework: topological spaces are data-types and continuous functions between topological spaces are computations. These interpretations form the basis for a systematic development of a propositional program logic from a denotational semantics.

We begin with considering predicates as subsets of an abstract set of states, and we study several semantic model of sequential languages. In particular we consider the weakest precondition and the weakest liberal precondition semantics. We relate them to three denotational models based on state transformation. The relationships between these axiomatic and denotational models generalize the duality of Plotkin between Dijkstra's predicate transformers and the Smyth powerdomain.

Then we extend sequential languages with specification constructs. We use the language of Back's refinement calculus which supports two kinds of unbounded non-determinism- Traditionally, the semantics of the refinement calculus is based on monotonic predicate transformers. Beside it, we give a denotational semantics based on state transformations, and an operational semantics based on a hyper transition system. We relate the three models as follows: the operational semantics coincides with the denotational semantics which, in turn, is dual to the predicate transformer semantics.

In order to study the semantics of concurrent languages, we refine the notion of predicates by considering affirmative predicates. They are open subsets of an abstract set of states equipped with a topology. This permits us to define dualities between the upper, lower and Vietoris powerspace constructions, and topological predicate transformers.

One of the above dualities is applied to prove the correctness of a new compositional predicate transformer semantics for a concurrent language. The semantics domain is a metric space which is shown to be isometric to the resumption domain of De Bakker and Zucker. Partial and total correctness, and also temporal properties are studied for this metric predicate transformer semantics.

Finally, we make an abstraction step by regarding predicates as elements of an abstract algebra. We consider a topological space as a function from the abstract set of affirmative predicates (with algebraic operations representing arbitrary unions and finite intersections) to the abstract

1571-0661/$ – see front matter © Elsevier Ltd
10.1016/S1571-0661(05)01613-0

set of specifications (with algebraic operations representing arbitrary unions and arbitrary intersections). We call this function an observation frame. We first show that topological spaces can be reconstructed from observation frames, and then we prove that observation frames are algebraic structures in a precise categorical sense.

The above theory is applied to extend the finitary domain logic of Abramsky to an infinitary one preserving completeness. As an example we extend Abramsky's finitary domain logic for transition systems to an infinitary logic with arbitrary conjunctions and arbitrary disjunctions. Our extension is conservative in the sense that the domain represented in logical form by the infinitary logic coincides with the domain represented in logical form by Abramsky's finitary logic.