POSSIBLE FUTURES, ACCEPTANCES, REFUSALS, AND COMMUNICATING PROCESSES

W.C. Rounds* and S.D. Brookes

Computer and Communication Sciences
University of Michigan

Programming Research Group
Oxford University

Abstract

Two distinct models for the notion of communicating processes are introduced, developed and related. The first, called the possible-futures model, is a generalization to nondeterministic systems of the familiar derivative (Nerode equivalence class) construction. The second, called the acceptance-refusals model, is a slight strengthening of a model introduced by Hoare, Brookes, and Roscoe. The PF model can be mapped onto the AR model homomorphically, and the equivalence classes of this map can be characterized by imposing a very natural equivalence relation on the PF model. The resulting quotient algebra admits a complete partial order structure in which the algebraic operations are continuous.

I. Introduction

We propose two mathematical models for the algebra of communicating sequential processes introduced by Hoare [H]. We think of CSP 'programs' as expressions denoting processes with concurrent and nondeterministic behavior, and assign meanings to these expressions using the methods both of formal language theory and denotational semantics. Starting with the primitive notions of event and sequence of events, we are able to define certain features of nondeterministic transition systems without having to introduce the notion of states. This makes it possible to introduce recursive expressions into the language and to assign them meanings as well.

Our first model is called the possible-futures model. It is intended to abstract a nondeterministic transition system, and was invented as the result of an attempt to define the so-called Nerode equivalence for nondeterministic systems. The second model, called the acceptance-refusals machine (ARM) model, was directly inspired by the refusal machines of Hoare, Brookes, and Roscoe [HBR]. ARMs are a slight strengthening of refusal machines and arose as the result of investigating the connections between refusal machines and possible futures. The results of that investigation will appear in a companion paper [BR].

*Research supported in part by sabbatical leave from University of Michigan and NSF Grant MCS-8102286.

The most interesting results from a theoretical point of view are, of course, the connections between the two models. We spend some time developing the first one, whose elements, called processes, are defined as certain relations on prefix-closed sets of strings. An algebra of processes is introduced based on the operators in CSP. In particular, we study the operations of autonomous choice, controllable choice, guarding, parallel composition, renaming, hiding, sequential composition, and inverse image. The possible-futures model, however, does not lend itself well to the general solution of recursive equations in these operators. A method is needed to introduce a partial ordering on processes which is complete in the sense of denotational semantics and for which the operations are continuous. This we do by defining an equivalence relation-called testable equivalence-on the possible-futures model, which is a congruence with respect to the algebra of processes, and which induces a (complete) partial order on the quotient space in a very natural way.

At this point, the ARM machines are introduced. We show that each equivalence class of testable equivalence is naturally represented by a machine. In fact, we demonstrate that testable equivalence is a congruence by constructing a homomorphism from the set of processes onto the set of machines. The algebra of machines is very much the same as that given by Hoare, Brookes, and Roscoe. This paper thus vindicates their definitions by showing how they can be derived from a more general model.

Several authors have attributed meanings to CSP ([FL], [FSP]). The present work, as well as that of Hoare, Brookes, and Roscoe, focuses on an abstract version of the language, which has no identifier conventions, labeling conventions, provision for variables, or scoping rules. We have attempted rather to make the mathematical semantics as universal and precise as possible, in hopes that it can be adapted to particular language designs.

We should also contrast these models with other models of parallelism in the theoretical literature.

We make a general distinction between the notion of a state-based and an event-based model. The former have been well-studied (see Keller [K] for a good representative) and are useful for many