ABADI & LAMPORT and STARK:  
towards a proof theory for stuttering, dense domains  
and refinement mappings  
(Extended Abstract)

Eduard Diepstraten†, Ruurd Kuiper‡  
Eindhoven University of Technology  
P.O. Box 513  
5600 MB Eindhoven, The Netherlands

Abstract

Crucial in proving refinement between specifications (of concurrent programs) is the role of  
ghost variables. On one hand they enhance expressivity. On the other hand they introduce  
stuttering and, in the case of refinement mappings, lead to the non-existence of such mappings.  
Semantically, the these problems are solved satisfactorily in the work of Abadi & Lamport  
[AL88]. Syntactically, however, their solutions have no obvious prooftheoretic counterpart.  
By formulating Abadi & Lamport’s concepts within Stark’s formalism for dense Linear Time  
Temporal Logic [Sta88] a step in this direction is made.

Keywords: Temporal logic, ghost variables, stuttering, refinement mappings, simulation rela-
tions, history and prophecy variables.

Contents

1 Desired abstraction level of descriptions  
1.1 Behaviour  
1.2 Satisfaction and refinement

2 Models  
2.1 Sequences  
2.2 Histories

3 Description languages  
3.1 \(\mathcal{L}(U)\) as a sequence logic  
3.2 \(\mathcal{L}(U)\) as a history logic

4 About expressivity  
4.1 Models for the enhanced logics

*Supported by the Netherlands Organisation for Scientific Research (NWO) under grant NF 62-519: Refinement and Education in Concurrent Systems (REX). E-mail address: wsined@win.tue.nl.

†Supported partially by ESPRIT project no. 3096 “SPEC: Formal Methods and Tools for the Development of Distributed and Real Time Systems”. E-mail address: wsinruur@win.tue.nl.
4.2 Problems in the enhanced logics
4.3 Can we still express refinement in a simple way?

5 Problems at the proof system level
5.1 Proving refinement via construction
5.2 Splitting safety and liveness
5.3 Stuttering problems again

6 Stronger proof rules

A Safety specifications
A.1 Relation between these approaches
A.2 Stark-specifications

B Counterexamples from [AL88]
B.1 Lack of historical information
B.2 Nondeterministic choices made too late
B.3 Lack of stutter steps

C AL-specifications in $L_G(U, t)$

D Proof rules of Abadi & Lamport
D.1 Refinement mappings
D.2 Addition of auxiliary (i.e. ghost) variables

Introduction

The new element in this paper is formulation of Lamport's Transition Axiom Method [Lam89] in a restriction of Stark's formalism for dense LTL. In this context two approaches are discussed: one based on Manna & Pnueli's discrete linear time temporal logic (LTL) [MP82], the other on dense LTL as developed in Stark's work on refinement [Sta88] and the work of Barringer et al. on fully abstract models for LTL [BKP86]. Technically speaking, we trace the causes of stuttering in mainly the discrete framework, more specifically the use of ghost variables to enhance the expressive power of these logics.

A rigorous framework is developed to describe various approaches to the so-called "stuttering problem". This problem is the following one:

1. The use of ghost variables in a specification enforces the existence of stutter steps (steps which leave the observable component of the state invariant).

2. An implementation (specification) that intuitively satisfies this specification may enforce less stuttering (stutter steps).

3. With obvious notions of satisfaction and refinement such an implementation will not formally satisfy the specification (because of lack of a sufficient amount of stutter steps).

4. In the context of refinement, the stuttering problem is: find a semantics and proof theory in which the above intuitive implementation indeed refines the original specification.