Reactive Spin and Promela

E. Najm & F. Olsen
ENST

October 12, 1995

Abstract

Reactive Promela is an extension to the Promela language which lets the user specify configurations of reactive automata. This provides a simple and powerful way to decompose a system.

To simulate and verify systems written in Reactive Promela the tool Reactive spin has been developed. It is a preprocessor for spin which translates a Reactive Promela system into a corresponding Promela system. The translated system can then be simulated and verified using spin. The main function performed by Reactive spin is to combine configurations of automata into Promela proctypes.

We demonstrate the language and the tool with the specification and verification of the HDLC protocol.

1 Introduction

When considering the problem of specification, decomposition is a central issue. Promela provides for two styles of composition of automata: loosely coupled (communication is by FIFO queues) and tightly coupled (communication is by rendezvous). A third style, the synchronous reactive style, has been widely advocated and used in the literature and in industry.

In the synchronous reactive style, a configuration of automata reacts to external events in a synchronous way: a collection of external events is treated thoroughly by the configuration before another event is taken and processed. In other words, the reactive configuration reacts to output events, and it is only at the end of the reaction that new inputs can be considered and processed. This kind of processing ensures that the speed of a reaction is higher than the delay between two consecutive input events.

The reactive style allows for powerful decomposition of specifications, beyond what is possible with merely rendezvous between automata. Furthermore, it reduces the state space explosion by constraining parallelism between automata.

Whereas Holzmann in [H91] proposes ways of reducing the complexity of systems (by incremental composition, minimisation, generalisation, atomic sequences, layering and structuring techniques, and so on), this is not a feature of the Promela language in itself. Instead it is a guideline for how to use the language for large, complex systems.

This paper describes an extension to Promela, whereby reactive processes ca be defined and instantiated. A reactive process is a configuration of synchronously composed automata. Besides the linguistic extension, this paper also describes a translation mechanism of reactive processes into Promela processes. This translation has been implemented in a preprocessor to spin, called rspin which translates a specification in Reactive Promela into an equivalent one in Promela.

References

[H91] G. Holzmann, "Design and validation of computer protocols", Prentice Hall Software Series, 1991.