Making Modelica Applicable for Formal Methods

Matthew Klenk Daniel G. Bobrow Johan de Kleer Bill Janssen
Palo Alto Research Center
3333 Coyote Hill Rd, Palo Alto, CA, 94304
{klenk,dekleer,bobrow,janssen}@parc.com

Engineers need to perform many different types of analyses as they design systems. Modelica has become a leading language to support numerical simulation. As a consequence there is widespread understanding of Modelica and a large number of Modelica model libraries available. This paper addresses the task of analyzing Modelica models with formal methods to derive system properties such as whether a design meets its requirements for all possible inputs. We report on our experience building a qualitative reasoner operating on Modelica models. In particular, we discuss the importance of leveraging the Modelica compiler to construct models for verification.

Modelica's reuse and flexibility are central to its appeal among designers, engineers, and researchers. Unfortunately, these features can also impede the application of formal methods to Modelica models. In this paper, we highlight five classes of problematic Modelica modeling practices:

- Artifacts for numeric simulation
- Unnecessary component model complexity
- Procedures
- Sequential states
- Incomplete models

For each class of modeling practices, we discuss a number of examples highlighting different ways in which the problem manifests. In each case, we seek to answer the following questions:

- Why do designers use it?
- Why is this difficult for formal methods?
- What should be done to enable formal verification?

We close the paper with a discussion of modeling principles to guide modelers in the future.