A translation of interaction relationships into SMV language

Zsolt Borsi

Department of Software Technology and Methodology Faculty of Informatics Etvs Lornd University

`bzsr@inf.elte.hu`

In this paper a translation of a particular scenario-based model into SMV language will be presented. SMV is the input language of the NuSMV model-checker tool[6]. Model checkers in general provide a verification way to prove that a given system meets its specification[4, 5]. By using model checking, errors of the system can be detected even in the very early phases of the software development process.

The Unified Modeling Language (UML) provides diagrams to describe the same system from different aspects. The notion of Interaction Overview Diagram (IOD) was introduced in the second version of UML for specifying the relationships between UML interaction diagrams and the control flow passing between them.

In this paper an algorythm for the translation of a hierarchical construct containing IODs will be presented. The top level diagram will be represented by the main module in SMV.

The idea of using model checking for verification is not new. There are various algorythms converting UML diagrams into SMV language[2, 3]. The novelty of this paper is, that the base of the translation are IODs. Moreover, the transition takes into account additional constructs which are not part of UML, but are used by various authors[1].

References

[1] J. Whittle, P.K. Jayaraman. Synthesizing hierarchical state machines from expressive scenario descriptions, *ACM Transactions on Software Engineering and Methodology (TOSEM), v.19 n.3, pp.1-45, 2010.*

[2] E.M. Clark, W. Heinle. Modular Translation of Statecharts to SMV, *Technical Report CMU-CS-00-XXX, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213., 2000.*

[3] R. Eshuis. Symbolic model checking of UML activity diagrams, *ACM Transactions on Software Engineering and Methodology (TOSEM), v.15 n.1, pp1-38, 2006.*

[4] E.M. Clark, O. Grumberg, D.A. Peled. Model Checking, *The MIT Press, Cambridge, MA, 2000.*

[5] O. Kupferman, M.Y. Vardi, P. Wolper. Module checking, *Information and Computation, 164(2), pp.322-344, 2001.*

[6] NuSMV Model Checker Home Page, *http://nusmv.fbk.eu*