The ObjectCheck Project

Fei Xie

(Dissertation Advisor: James C. Browne)

Dept. of Computer Sciences, Univ. of Texas at Austin, Austin, TX 78712, USA
Email: {feixie, browne}@cs.utexas.edu Fax: +1 (512) 471-8885

1 Overview

This abstract introduces the ObjectCheck project, summarizes its current status, and discusses its future directions. In this project, we have designed and implemented model checking support [1] for executable object-oriented software system designs specified in xUML [2], an executable dialect of UML. To alleviate the state space explosion problem, we have introduced and implemented an integrated state space reduction framework [3] for model checking executable object-oriented software systems designs. Currently, we are exploring the synergism [4] between component-based development (CBD) and model checking, which enables building reliable component-based software systems and reduces complexity of verifying these systems by utilizing their compositional structures.

2 Model Checking Support for xUML

The basic model checking support for executable object-oriented software system designs specified in xUML can be summarized as follow:

- A system design is specified in xUML as an executable model.
- A property to be checked on the design is specified in an xUML level logic.
- The xUML model and the property are automatically translated to a model and a query in the S/R [5] automaton language.
- The S/R query is checked on the S/R model by COSPAN [5] model checker.
- If the query fails, an error track is generated by COSPAN and is automatically translated to an error report in the name space of the xUML model.

The S/R automaton language employs synchronous parallel execution semantics and variable-sharing communication paradigm. Therefore, we simulate the asynchronous interleaving message-passing computation model of xUML in the synchronous parallel variable-sharing computation model of S/R.

3 Integrated State Space Reduction Framework

To alleviate the state space explosion problem, a general framework for integrated state space reduction in model checking executable object-oriented software system designs, has been established. The framework structures the application of state space reduction algorithms into three phases with different algorithms applied in each phase. The interactions between these algorithms are

explored to maximize the aggregate effect of state space reduction. Automation support for the framework has been proposed and partially implemented. The framework is presented for system designs modeled in xUML, but can also be used to structure integrated state space reduction for other representations. To further improve the applicability of the framework, domain-specific design patterns can be explored to instantiate the framework for different application domains. An instantiation [3] of the framework for distributed transaction systems has been defined and its partial implementation has been applied to the design model of an online ticket sale system. The dimension of software system designs that are model checkable was found to be greatly extended.

4 Verification of Component-based Software Systems

We have proposed an approach to integrating model checking into the CBD of software systems. In this approach, temporal properties of a software component are specified, verified, and packaged with the component. Selection of a component for reuse considers not only its functionality, but also its temporal properties. When a component is composed from simpler components, temporal properties of the composed component are model checked on its abstraction constructed based on the composition and the properties of its sub-components. Model checking enhances the reliability of software systems constructed with CBD and the compositional structures of these systems significantly reduce the complexities of model checking. This approach has been applied to enhance the reliability of run-time images of TinyOS [6], a component-based run-time environment for networked sensors.

5 Conclusions and Future work

The ObjectCheck project provides a testbed for new model translation techniques, state space reduction algorithms, and software development approaches. More state space reduction algorithms will be included to enhance its model checking capabilities and more case studies on developing reliable software systems will be conducted to validate its existing capabilities.

References

1. Xie, F., Levin, V., Browne, J.C.: Model Checking for an Executable Subset of UML. Proc. of 16th Inter. Conf. on Automated Software Engineering (2001)
2. Project Tech.: http://www.projtech.com/pubs/xuml.html. Project Tech. (2001)
3. Xie, F., Browne, J.C.: Integrated State Space Reduction for Model Checking Executable Object-oriented Software System Designs. Proc. of FASE 2002 (2002)
4. Xie, F., Browne, J.C.: Verified Systems by Composition from Verified Components. UTCS Technical Report TR-02-40 (2002)
5. Hardin, R.H., Har'El, Z., Kurshan, R.P.: COSPAN. Proc. of 8th International Conf. on Computer Aided Verification (1996)
6. Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D., Pister, K.: System Architecture Directions for Networked Sensors. Proc. of ASPLOS-IX (2000)